12-14-2016

Content Dissemination in Mobile Social Networks

Chenguang Kong

Content Dissemination in Mobile Social Networks

by

Chenguang Kong

Under the Direction of Xiaojun Cao, PhD

ABSTRACT

Mobile social networking(MSN) has emerged as an effective platform for social network users to pervasively disseminate the contents such as news, tips, book information, music, video and so on. In content dissemination, mobile social network users receive content or information from their friends, acquaintances or neighbors, and selectively forward the content or information to others. The content generators and receivers have different motivation and requirements to disseminate the contents according to the properties of the contents, which makes it a challenging and meaningful problem to effectively disseminate the content to the appropriate users.

In this dissertation, the typical content dissemination scenarios in MSNs are investigated. According to the content properties, the corresponding user requirements are analyzed. First, a Bayesian framework is formulated to model the factors that influence users behavior on streaming video dissemination. An effective dissemination path detection algorithm is derived to detect the reliable and efficient video transmission paths. Second, the authorized content is investigated. We analyze the characteristics of the authorized content, and model the dissemination problem as a new graph problem, namely, Maximum Weighted Connected

subgraph with node Quota (MWCQ), and propose two effective algorithms to solve it. Third, the authorized content dissemination problem in Opportunistic Social Networks(OSNs) is studied, based on the prediction of social connection pattern. We then analyze the influence of social connections on the content acquirement, and propose a novel approach, User Set Selection(USS) algorithm, to help social users to achieve fast and accurate content acquirement through social connections.

INDEX WORDS: Content Dissemination, Mobile Social Networks, Opportunistic Social Networks

CONTENT DISSEMINATION IN MOBILE SOCIAL NETWORKS

by

Chenguang Kong

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2016

Content Dissemination in Mobile Social Networks

by

Chenguang Kong

Committee Chair: Xiaojun Cao

Committee: Raj Sunderraman

Anu Bourgeois

Yi Zhao

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

Dec 2016

DEDICATION

This dissertation is dedicated to my family.

ACKNOWLEDGEMENTS

First of all, I shall thank my supervisor Dr. Xiaojun Cao for his invaluable and continuous guidance, support and supervision throughout my study. He taught me how to be a researcher, and lead me to start my work. It is him who educate me to a mature PhD graduate from a raw recruit. He is always acute and to-the-point when identifying a research problem, and guides me to the correct directions. He taught me a lot in research methodologies and working attitude, which are far more important than the knowledge learned from courses or books.

Second of all, I would like to thank my committee members for their help and effort on the reviewing and advertising to the completeness of my dissertation.

Third of all, I wish to thank my friends for their emotional support, and technical help and suggestion when I carried out my experiments I also appreciate the support and help from the Department of Computer Science by providing countless and valuable resources, courses, opportunity and so on

Last but not least, I owe a lot to my dear parents and all the other family members, for their encouragement and support, throughout my life.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . . . **v**

LIST OF TABLES . . . . . **vii**

LIST OF FIGURES . . . . . **viii**

Chapter 1 INTRODUCTION **1**

1.1 Content Dissemination in MSNs 1

1.2 Objectives of Content Dissemination 3

1.3 Challenges and Contributions 4

Chapter 2 LITERATURE REVIEW **7**

2.1 Network Connectivity on Content Dissemination 7

2.1.1 Centrality 8

2.1.2 Community 9

2.1.3 Opportunistic Connections 11

2.2 User on Content Dissemination 12

2.2.1 Interests 14

2.2.2 User Influence 15

2.2.3 Mobility 17

2.2.4 Privacy and Security 18

2.2.5 Incentive 18

2.3 Content Perspective 20

2.3.1 Simple Content 20

2.3.2 Streaming Content 21

Chapter 3 BAYESIAN-BASED CONTENT DISSEMINATION FRAMEWORK 23

3.1 The Framework for Video Sharing in MSN 24

3.1.1 Abstract Distribution 25

3.1.2 Bayesian Model 26

3.1.3 Video Request and Transmission 33

3.1.4 Multiple Streaming Transmission 33

3.2 Simulation and Performance Analysis 34

3.2.1 Performance Evaluation in Tree and Grid Networks 35

3.2.2 Performance Evaluation in Random Networks 38

Chapter 4 SEMI-CONTROLLED CONTENT DISSEMINATION . 43

4.1 Maximum Weighted Connected subgraph with node Quota (MWCQ) 45

4.1.1 Weight Calculation 46

4.1.2 MWCQ Problem Formulations 47

4.2 Dynamic Programming based SAID (DP-SAID) Algorithm . . . 49

4.3 Two-Hop based greedy SAID (TH-SAID) Algorithm 58

4.4 Performance Evaluation 59

4.4.1 Network Setting 59

4.4.2 The Sum of Weight Performance 59

4.4.3 Running time 60

4.4.4 Impact of Network Structure 61

Chapter 5 CONTENT DISSEMINATION IN OPPORTUNISTIC SOCIAL NETWORKS 64

5.1 Authorized Content Dissemination in IOSNs 65

5.2 Social Connection Pattern 68

5.3 SCP Based Content Dissemination Algorithm 70
5.4 Simulation and Evaluation 74

Chapter 6 USER RECOMMENDATION FOR EFFICIENT CONTENT ACQUIREMENT **83**

6.1 Social Connection optimization for Efficient Information Acquisition 85
6.1.1 Information Correlation 86
6.1.2 Timeliness Calculation 87
6.1.3 Accuracy Calculation 88
6.1.4 MapReduce-based Information Processing 90
6.1.5 Social Connection Optimization for Efficient Information Acquisition(SCOIA) 92
6.2 User Set Selection (USS) Algorithm 93
6.3 Performance Evaluation 94
6.3.1 Spam Ratio Threshold 96
6.3.2 Timeliness Threshold 97
6.3.3 Algorithms Comparison 98

Chapter 7 CONCLUSION **99**

REFERENCES **101**

LIST OF TABLES

Table 2.1 Summary of connection properties 8
Table 2.2 Summary of Users Properties 13
Table 3.1 Parameter setting used in simulation 35
Table 4.1 The process of DP-SAID algorithm 55
Table 4.2 Network Setting 59
Table 5.1 Information about the OSN datasets 75
Table 6.1 Notations used in SCOIA 86

LIST OF FIGURES

Figure 3.1 Video sharing in MSN 24

Figure 3.2 Scheme of the abstract distribution flow 26

Figure 3.3 Example for multi-path transmission 27

Figure 3.4 Video request probability (VRP) distribution in a tree network . . 36

Figure 3.5 Video request probability (VRP) distribution in a grid network . 37

Figure 3.6 Video request probability distribution in a random network 40

Figure 3.7 Video request probability distribution in random networks with different node degree 41

Figure 3.8 Single-path transmission 42

Figure 3.9 Multi-path transmission 42

Figure 4.1 An example of SAID flow 45

Figure 4.2 An example of DP-SAID algorithm 52

Figure 4.3 An example of shared path structure with $R = 8$ 53

Figure 4.4 A Network structure example of the lower bound case 54

Figure 4.5 Network with $N = 100$, $D = 10$ 62

Figure 4.6 Network with $N = 500$, $D = 20$ 62

Figure 4.7 Network with $N = 1000$, $D = 50$ 62

Figure 4.8 Running time with $N = 100$, $D = 10$ 62

Figure 4.9 Running time of network with $N = 500$, $D = 20$ 62

Figure 4.10 Running time of network with $N = 1000$, $D = 50$ 62

Figure 4.11 DP-SAID on network with $N = 1000$ 63

Figure 4.12 TH-SID on network with $N = 1000$ 63

Figure 4.13 Running time of DP-SAID on network with $N = 1000$ 63

Figure 4.14 Running time of TH-SAID on network with $N = 1000$ 63

Figure 4.15 Average node degree of the subgraph obtained by algorithms . . . 63

Figure 5.1 An example of authorized content dissemination in IOSNs 67
Figure 5.2 Expected reward change during the content dissemination 71
Figure 5.3 Dissemination results of Infocom dataset 75
Figure 5.4 Dissemination results of Sigcomm dataset 76
Figure 5.5 Dissemination results of Infocom dataset on training length 77
Figure 5.6 Dissemination results of Sigcomm dataset on training length . . . 78
Figure 5.7 Distributions of the received content copies of Infocom Dataset . . 79
Figure 5.8 Distributions of the received content copies of Sigcomm Dataset . 80
Figure 6.1 An example of support ratio calculation 89
Figure 6.2 The nested MapReduce algorithm 91
Figure 6.3 Support ratio of selected users 95
Figure 6.4 Number of attractive information offered 95
Figure 6.5 Average support ratio of selected connections 95
Figure 6.6 Average number of attractive information offered 96
Figure 6.7 Average support ratio of selected connections 96
Figure 6.8 Average number of attractive information offered by selected connections 96

Chapter 1

INTRODUCTION

1.1 Content Dissemination in MSNs

Social Networks is a type of networks that are constructed by the communication and connections of the social networks users. In recent years, Social networks have received incredible growth. Studies find that there are over 1.79 billions active social network users in 2014, as more than 64 percent of internet users[1]. In social networks, the connections(or edges) in networks represent possible social communication links, such as friend-to-friend communication, encounter between strangers and so on. Based on different social communication connections, various social network services have been provided to social networks users by social service providers, such as Facebook[2], Twitter[3], Instagram[4]. Through those social network service, people enjoy a series of convenient and attractive communication and interaction experience. They can find friends with similar interests, share and acquire personal photos and blogs, supply to or ask for advice and so on.

Among those social network service, the content dissemination is one of the most pervasive and attractive feature. In the social network based content dissemination, the social network users generate different types of contents, such as news, messages, music and movies. The content is disseminated by the content generators to their social network connections. Those social network connections then will help to disseminate the content to more users in ad hoc mode.

The content dissemination applications are promoted by the development of wireless technology and the mobile devices. With the advancing wireless technologies such as Wi-Fi[5], WiMAX[6] and 4G LTE[7], Mobile Social Networks (MSNs) provide much more flexible and ubiquitously accessible wireless connection and communication among social network users. The usages of mobile devices in MSNs makes it possible for social network users to

communicate and interact with others at anytime and any location. In MSNs, social users are allowed to generate timely and location based contents through their mobile devices, such as local news, traffic information, tourist videos, events happening around them, or restaurant information they just visit. Through MSNs, users are motivated to diffuse those contents to other social network users in neighbor.

Content dissemination is a meaningful application in Mobile Social Networks(MSNs), which can efficiently improve the content sharing and spreading. Though there are many different forms of content dissemination, like cellular network based client-server framework, subscribe/publish framework, the ad hoc netowrk based content dissemination appears much more interesting and challenging contribution, which utilize the pairwise communication of MSN users. Mobile social users will communicate with each other if and only if there exit mobile social connections, which means: (1) mobile social users locate within the transmission range of each other. The transmission range is decided by the communication technologies used by the mobile devices of mobile social users(e.g., around 100 meters for typical 2.4 GHz WiFi communication); (2) there exist certain social relationships attached to the connections between the communicating users. The social relationships considered in mobile social networks include multiple aspects as friendship, colleagues, users with common interests, strangers and so on. The social relationships play am important role in many content dissemination applications.

In this paper, we investigate the problems and challenges involving ad hoc network based content dissemination in MSNs, and discuss the possible solution on this problem. Generally, the content to be disseminated is generated by content providers(CPs), and diffused to the nearby social connections within the communication range of content providers. When a user receives the content from its social connections, it will take actions on those content. There are various actions that may be taken by MSN users, such as save the content into buffering, discard the content, forward the content to other social connections, and so on. If the content is forwarded by a user to its social connections, the dissemination process is continued until no more forwarding action happens.

1.2 Objectives of Content Dissemination

There are different objectives existing during the content dissemination process. First important objective to maximize the delivery rate. The delivery rate is calculated as the ratio of the number of content receivers over the total number of mobile social users in networks. This objective is significant for information diffusion. By maximizing the delivery rate, the content can be received and known by as many users as possible. To achieve this objective, epidemic flooding[8] may be an effective approach, though it may cause redundant caching and traffic. CEDO[9] studies to maximize the total delivery rate under the constraints that users have limited resources to store the content.

Second, it is meaningful to minimize the dissemination time. The dissemination time measures the time used for the dissemination process. Particular type of content, such as news and alert, requires that the content disseminated to receivers as soon as possible to ensure the effectiveness of the content. For instance, Lu et. al.[10] and Chen et. al.[11] try to minimize the dissemination time and maximize the dissemination speed by choosing the most influence users in the networks.

Third, mobile social users may have different importance to the networks or content providers. Hence, it is necessary to disseminate the content to most important or valuable users. One important factors determining the importance is users interests on the content. For example, users interested in content are more active during content dissemination process. Therefore, some content dissemination works aim to disseminate the content to interesters as much as possible. The work [12] studies to maximize the total weight of the content receivers, which is in proportion to how users are interested in the content. To achieve that objective, we need consider not only the interest of users themselves, but also the capacity that users connect to other interested users.

Last but not least, the concern on privacy and security requires that the content dissemination process should involves as few unnecessary users as possible. Therefore, it is one objective to minimize the number of users involved. Gao et.al. in[13] aim to maximum the cumulative cost effectiveness, which is defined as the ratio of the number of content inter-

esters who receive the content upon the total number of relays for the contents. To achieve the objective, the authors propose to disseminate content through interesters as much as possible, and intently select non-interesters as rely users to help to forward the content. Another instance of such work is [14], in which the content dissemination network is built based on the human connectivity network and the interest network.

The content dissemination performance in MSNs is influenced by series of factors. the most significant among them can be categorized into three aspects: network connectivity, user preference and content. The network connectivity on content dissemination describes the connectivity properties of the networks and users. For instance, the capacity of a user connects to others in the network determines whether this user is suitable to help disseminate the content to other users. User preference influences users' possible behaviors during dissemination process. For example, a user is probably to accept the content matching its personal interests and reject the others. Hence, the analysis on user preference and its influence on user behavior plays an important role to develop efficient dissemination scheme. The content dissemination process should consider the attributes and requirements of content as well. To disseminate streaming content, it is necessary to ensure the duration and bandwidth of the connection, so that the streaming content can be transmitted smoothly and completely.

1.3 Challenges and Contributions

Given the objectives of content dissemination and the characteristic of the mobile social networks, the content dissemination process present several major challenges in different scenarios.

(i) The difference and complicated attributes of the social connections make it a significant and challenging work to disseminate content based on the analysis of mobile social connections. In most scenarios, the mobile social connections are heterogeneous, and have different importance and influence on content dissemination performance. Hence, the first challenges of content dissemination lies how to analyze and utilize the heterogeneous mobile

social connections.

(ii) As one of the most important component in mobile social networks, mobile social users have their own concern and requirements. For example, social users have distinctly attitudes and preferences to the contents disseminated in MSNs. They show strong interest in the contents which provide interesting or useful information to them, and are willing to contribute the physical resource of their mobile devices to participate in the dissemination process of the contents. However, most social users will ignore or discard the uninteresting contents they received, which will damage the dissemination process. Hence, to effectively disseminate content in MSNs, we must analyze users attribute and concerns that may influence the dissemination performance.

(iii) In many scenarios, content itself contains kind of attributes, some of which impose particular requirement on the dissemination process. For instance, the video content has high requirement on the mobile social connection duration and bandwidth; privacy sensitive content dissemination relies on the privacy aware dissemination protocols. As a result, a content dissemination framework need to meet the requirement and challenges of content as well.

Focusing on the challenges of content dissemination in mobile social networks, we have investigated the characteristics of the content disseminated in MSNs. The user interests and influence are analyzed to model the behavior pattern of the social users during the dissemination. Furthermore, we have proposed effective and efficient communication and dissemination protocols to satisfy the demands of both content requesters and content providers. The major contribution of this dissertation can be summarized as follows.

(i) We investigate the dissemination problem for large size of contents such as streaming videos, which require high transmission bandwidth and stable connections. We categorize the factors of network and social behaviors in video content dissemination. Three important categories including neighbor confidence, interest matching, and physical network resources are considered. We then develop a new Bayesian network model to facilitate the process of video sharing in mobile social networks, and propose a new framework to effectively distribute

video data based on the proposed Bayesian network model. The video request probability distribution in the Bayesian model is used as the primary parameter for the routing decision.

(ii) The features and challenges of authorized contents are analyzed, which have strictly constraints on content copy and editing and may generate benefit/reward to content generator. We study the authorized content dissemination problem to maximize the reward obtained by content generator and form the Maximum Weighted Connected subgraph with node Quota (MWCQ) problem. Two efficient heuristic algorithms, Dynamic Programming based SAID (DP-SAID) and Two-Hop based greedy SAID (THSAID) algorithms, are derived to provide either accurate or low cost computing solution for the problem.

(iii) The Interest-centric Opportunistic Social Networks (IOSNs), in which users move around for the activities or locations they are interested in, is also studied in this dissertation. Upon the content dissemination problem in IOSNs, we propose the Social Connection Pattern (SCP) to describe the interest distributions of users's social connections. We then develop the Social Connection Pattern based Dissemination (SCPD) algorithm to identify a proper content dissemination strategy when two users contact.

(iv) We investigate the influence of social connections on the content dissemination, and propose to enhance the content dissemination performance by recommending appropriate social connections. We analyze the accuracy and timeliness performance provided by social connections, and propose our model to measure them. Then social connections are recommended to optimize both the accuracy and timeliness of the content disseminated.

The remainder of this dissertation is organized as follows:

We discuss the literature review in Part 2. Then the work on Bayesian networks based streaming video dissemination framework is studied in Part 3. Part 4 presents the semi-controlled dissemination algorithm on authorized content. My work on the authorized content dissemination problem in OSNs is present in Part 5. Part 6 discusses the user recommendation problem on information acquirement. Last, we conclude the proposal in Part 7.

Chapter 2

LITERATURE REVIEW

Recently content dissemination has drawn great attention from both academia and industry. In this chapter, we mainly categorize and evaluate the related work.

2.1 Network Connectivity on Content Dissemination

To efficiently disseminate the content in mobile social networks, the first question is how to detect the dissemination paths from the content providers to every receivers. This question could be answered by identifying the users who can support the dissemination process most(e.g., the users connecting the most other users, the users performs the lest dissemination delay, etc.). In most cases, the capacity that a user support the dissemination process highly relies on the network connectivity properties. The network connectivity describes how the network is connected and the corresponding properties of network users and connections. Whether a social connection can be selected on the dissemination path is highly relative to the quality of service provided by that social connection. The quality of service of a social connection is influenced by the properties of the social connections, users' attitude and possible behavior, user's communication pattern and so on. To effectively disseminate contents in the network, we need to analyze the connection quality of service and detect the most effective and reliable dissemination path to targeted receivers.

In this section, we classify state-of-the-art according to the network connection properties focused. The connections present difference properties on duration, frequency, reliability, etc. For instance, the connections between two close friend would be much more frequent and reliable than the connection between two strangers. Therefore, when the content is disseminated through mobile social networks, the properties of the connections should be taken into consideration. Table 2.1 lists the importance properties of connections that have

Table 2.1. Summary of connection properties

Connection Properties	Characteristics	Topic Categories	Related works
Degree Centrality	Measure the number of users who are the connection of a user	Measure and utilize centrality to help disseminate content	[15] [16]
Closeness Centrality	Measure the distance between a user and the other users in the network		
Betweenness Centrality	Measure the shortest paths in the network traveling trough a user		
Community	Formed by group of users who have high frequency and long duration to communicate with each other	Community detection	[17][18] [19]
		Community based content dissemination	[20] [21]
Opportunistic Connection	Unstable Connections between users which are highly dynamic and non-deterministic	Opportunistic connection based content dissemination	BubbleRap[22], [23]

significant impact on the process of content dissemination in mobile social networks.

2.1.1 Centrality

Centrality is one of the basic connection properties that have been widely studied. Centrality is measured to indicate the topological importance of a user within the network [15] [16]. A central node typically has a stronger capacity to connect to other users in the network, and hence may play a more importance role during the content dissemination. The centrality of a user can be defined in several ways, including degree centrality, closeness centrality and betweenness centrality. Degree centrality measures the number of direct connections (or neighbors) involving a user, which is calculated as:

$$C_D(v) = deg(v) \tag{2.1}$$

. User with high degree centrality can be thought as popular users with a large number of direct connections or neighbors, hence they are more helpful to diffuse content to others.

Closeness centrality of a user is calculated as the reciprocal of its average shortest distance to all other users in the network. The Closeness centrality is calculated as:

$$C_C(v) = \frac{1}{\sum_{u \in V} d(u, v)} \tag{2.2}$$

where $d(u, v)$ is the shortest distance between user u and user v, V is the user set of the network. Therefore, the higher the closeness centrality of a user v is the shorter user v is from other users.

Betweenness centrality measures the importance of a user on the network connectivity among other users. The betweenness centrality is quantified as the number of shortest paths in the whole network traveling through the user.

$$C_B(v) = \sum_{s,t \in V} \frac{n_{st}(v)}{n_{st}} \tag{2.3}$$

in which n_{st} is the number of shortest paths from user s to user v while $n_{st}(v)$ is the number of those paths that travels through v.

The centrality of users can be utilized to help detect the routing to destination in information dissemination. Daly et. al. [15] propose to send the information to users with higher betweenness and similarity to destinations when two users encounter.

2.1.2 Community

The social network users who connect to each other with high frequency and duration form communities in mobile social networks. The communities identified in Mobile social networks can be used to make smarter dissemination decision comparing to random flooding. There are multiple approaches having been proposed to detect the communities in the mobile social networks, including modularity based approaches, betweenness centrality based approaches and so on. Modularity is defined as the fraction of connections in a network that

connect users within communities minus the expected value of the same quantity in a graph with the same communities but random connections between the users. The work in [18] detects communities by the Louvain method, in which each user forms its own community initially. Then in each step a user is added to the community which maximally increase the local modularity gain.

The Max-min modularity is a community structure detection methodology proposed in [17], which takes the connection pair as a positive sign of a strong community structure while separating disconnection pair as either possible related relationship or unrelated relationship. Hence the community division criteria is set as maximizing the number of edges between groups and minimizing the number of unrelated pairs within groups.

The community detection can also be completed based the betweenness centrality, as proposed in [19]. In the betweenness centrality based community detection algorithm, the betweenness of all connection in the network is calculated. At each step, the edge with the highest betweenness is removed and the betweenness of all connections affected by the removal is recalculated. The algorithm terminates until no edges remain or the community size meets the requirement. By utilizing the properties of betweenness, this method can efficiently detect the communities that formed by the network connections. However, the requirment to update the betweenness of nodes after each edge removing increases the computation cost of this method. Other community detection algorithms include Label propagation[24] and Communities from edge structure and node attributes(CESNA)[25].

The detected community can be utilized to improve the content dissemination process. Xiao et.at.[20] propose a community aware routing scheme in MSN. Two routing phases are defined in this scheme. In the initialization phase, the network with V users is simplified into a virtual network with L communities based on the community detection algorithm. Then in the routing phase, routing decision is made based on the simplified small size networks to minimize expected delivery delays.

Based on the property in the communities that the connection frequency between community members is much higher than that between strangers, the authors in [21] propose

to minimize the network wide provisioning cost by efficiently caching the content and increasing the sharing among MANETs. In this work, the cashing space is split into three component. The duplicate caching area is used to store the very popular objects which will be shared across and outside the communities. The unique in community caching space is reserved for the cooperation among in-community users. The content stored in this space is unique in the community, and shared to all community member. The last caching space is unique in network, which helps users to cooperate with strangers. To minimize the network cooperation cost, the best split factors are determined based on the encounter frequency of users with communities and across communities.

2.1.3 Opportunistic Connections

In many scenario, the connections between users are opportunistic, in which the connections have short duration and disconnect frequently. In addition, the connections are non-deterministic because of the mobility of users. The network with opportunistic connections is called opportunistic social network (OSN). In the opportunistic social networks, users connect to and communicate with other users who are within the communication range at certain time. The communication has high probability to terminate after a short duration.

To effectively disseminate content in opportunistic social networks, the dissemination decision should be made by every user having content to disseminate when he has a new opportunistic connection. The decision should be made based on the answers of two questions: (i) what is the possible future connections of the new opportunistic connection; (ii) how the opportunistic connected user can help to achieve the dissemination objective by further disseminating the received content to other users in future.

BubbleRap[22] is a work which tries to answer the questions by analyzing the centrality and community structures in opportunistic social networks. In BubbleRap scheme, users have both global rankings and local rankings. The global ranking describes the population of a user in the whole network(e.g., the degree centrality). The local ranking denotes how popular a user is in its own community. To forward content from a source user node to a

destination user node, the BubbleRap scheme first bubbles the content up the hierarchical ranking tree according to the global ranking of users, until the content is forwarded to a user within the same community as the destination user. Afterwards, the content will be forwarded within the community by using the local ranking tree until the destination user is reached or the content expires. The BubbleRap scheme does not require every user a global knowledge of the ranking. Instead, users just need to compare the ranking with the other users they encounter.

The content diffusion in OSNs is studied in [23]. This work considers the problems in content diffusion of both contact probability and content propagation order. To select the buffering scheduling between users and content, [23] utilizes both the friendship among users and the homophily phenomenon, which describes that friends usually share more common interests than strangers. Hence, [23] proposes a content diffusion strategy that diffuses the most similar content between friends and the most different content between strangers. In detail, if a user encounters a friend connection, it first diffuses the most similar content of their common interests to its friends first. Otherwise, if the new connection is a stranger, it chooses the propagation order of the most different content from their common interests first. It is shown that this content diffusion schemes performs better diffusion speed and content access delay .

2.2 User on Content Dissemination

Users' preference properties, such as interests and motivation, highly influences their possible behaviors acted during content dissemination. The influence of users' preference properties lies mainly in the aspects as follows. First, users' interests on the content they receive varies, which brings different strategy when processing the received content. For the content they are interested in, they would have stronger motivation to participate in the content dissemination process by maintaining longer connection duration and spending more physical resources such as bandwidth and storages. Nevertheless, less attention and support may be supplied for the content they are uninterested in, which cause a higher

Table 2.2. Summary of Users Properties

User Preference Properties	Characteristics	Problem Categories	Related Works
Interests	Describes users' interests on the disseminated content	How to satisfy users' interests concern	Gao[13], ContentPlace[26]
		How to utilize interest driven behaviors	Onside[27]
User Influence	Describe how users are influenced by friends, strangers and others	user influence model	LT model and IC model [28], time influence model[29]
		content dissemination with consideration of user influence	[11]
Incentive	Describe how users are motivated to participate content dissemination	Incentive scheme	Give2Get[30], Tit-for-Tat[31],credit[32]
Privacy and security	Describes users' demand on the privacy and security involved in content dissemination	Disseminate content through privacy ensured communication	Whisper [33], MCONs[34], [35]

probability to discard those content. Secondly, users mobility has directly influence on the network connections and further the dissemination paths for the content delivery. Third, users' preference on the privacy and selfishness will restrict the possible solution of content dissemination in mobile social networks. Because of the privacy concern or selfishness, users may choose to reject to help disseminate the content. Table 2.2 summarizes the properties of users that have high influence on the performance of content dissemination in mobile social networks. Hence, the concern and requirements from users need to be satisfied when a content dissemination scheme is proposed. Researchers have studied how to achieve the content dissemination objectives and improve the dissemination efficiency by analyzing and accomplishing users preference. In this section, we summarize those works according to the

user preference perspective.

2.2.1 Interests

Disseminating content based on users' interests on the content is a pervasive approach since users interested in the content have higher probability to contribute to the dissemination process, by maintaining connections, contributing memory for content storing and so on. Most content dissemination works based on user interests analyze users' behaviors influenced by users' interests on the content.

The influence of users interests on content dissemination lies on various aspects. First of all, users interested in the content are the desired receiver in many cases. Users are self-motivated to receive the content that they are interested in, and would like to participate the dissemination process to receive the interesting content. Second, users with different interests on the content may have distinct performance upon the content dissemination behavior. For example, it's regular that users spend their limited buffer on the interesting content comparing to others. Users with similar interests also have high frequency and probability to communicate with each other, which can provide highly efficient dissemination process.

To analyzing and further utilizing the influence of user interest on content dissemination process, there are two major directions focused by researchers. First, users concern on personal interests should be satisfied during the content dissemination. Users may not would like to be disturbed by uninterested content, in which the content should be disseminated and transmitted among the interested users only. Besides, the limited buffer of users may force users to choose an efficient caching strategy so that they can cache the interested content as much as possible. Targeting on these problems, Gao et.al.[13] propose a user centric dissemination approach. The objective of this approach is to maximum the cumulative cost effectiveness, which is defined as the ratio of the number of content interesters who receive the content upon the total number of relays for the contents. To achieve the objective, Gao et.al. divide the dissemination process into two parts: uncontrolled dissemination part and controlled dissemination part. In the uncontrolled dissemination part, the content is

disseminated among the interested users automatically without help of additional relays. Then in the controllable part, a number of relays are intentionally selected among the non-interested users according to their capabilities of forwarding content to interested users.

ContentPlace[26] focuses on the problem of buffering strategies, and proposes to organize the content into different channels, to which MSNs users subscribe according to their interests. In ContentPlace, when two users contact, they advertises the content they are interested in to each other. In addition, the content currently carried by them is also exchanged. To optimize the content dissemination objective, ContentPlace defines a utility function , according to which each user can associate a utility value to any content. Hence, during the encounter with another user, a user computes the utility values of all content stored in both local cache, and adjust its local cache to maximize the utility value.

Another types of research that have been done by researches is to utilize the interests of users to improve the content dissemination process. Driven by interests, users show particular interesting phenomena that can be utilized to enhance content dissemination. Onside[27] takes advantages of the fact that users with common interests tend to meet each other with higher frequency. In Onside, a user will download the content from an encounter if and only if the topic of the content is self interested, friends interested or encounter interested.

2.2.2 User Influence

in social networks, users are influenced by other users. For instance, a user may be interested in some content because of others recommendation. In addition, the information received from others may inspire particular actions. The user influence plays an important role in content dissemination as it directly impacts users possible behavior.

Corresponding to different scenarios, there are different influence models shown. Most of those influence models are developed on probabilistic model, in which a user has a probability to influence his neighbors/connectors. The probability is determined by multiple factors, including the social connection strength, friendship and other social relationships, connection history, influence history, time factors and so on.

Two basic influence models are discussed in [28]: Linear Threshold(LT) model and Independent Cascade(IC) model. In LT model, each connection has a weight and each user has an influence threshold. A user becomes activated if the weighted sum of its active neighbors exceeds its threshold. In IC model, each connection has an activation probability. Influence is propagated by activated users independently activating their inactive neighbors, based on the connection activation probabilities.

Based on the basic influence models, researchers have developed other influence model by considering additional factor such as time. the authors in [29] propose a continuous time model and a discrete time model. In the continuous time model, the probability of influencing depends on time. With time increasing, the probability of influence decays, following an exponential decay model. Discrete time model is an approximation of continuous time models while providing little testing cost. In discrete time model, the probability of user v influencing user u at a time window after v performs the action is constant.

The content dissemination process can be mapped as the influence propagation process in many scenarios. When a user disseminates content to another user, it can be analyzed as the influence action between two users. If a user accept the content, that means the user is activated by other users; otherwise, the user keeps deactivated. Hence, the content dissemination problem can be solved by user influence analysis. The authors in [11] propose to maximize the influence spread by selecting the seed nodes. A local directed acyclic graph(DAG) is constructed surrounding every node v in the network. Rooted at v, the DAG covers a significant portion of influence propagation in which the influence from seed to user v is only propagated within the local DAG of v. After that, the seeds that provide the maximum incremental influence spread can be selected with a greedy approach.

The user influence may be represented as the strength of social connection as well. For example, a friend typically has stronger influence on users behavior than a stranger. In other words, a friend has a higher social connection strength than a stranger. Hence, we can model the user influence as the strength of the social connections. Wang et. al.[8] propose a distributed social tie strength calculation mechanism to identify the relationship of

social connections. Based on the tie strength, the dissemination process can be executed in two phase: weak tie-driven forwarding and strong tie-driven forwarding. In weak tie-driven forwarding, the content is forwarded though weak social ties to local bridges users. After the content has been propagated to communities, the strong social ties will be utilized to disseminate content through influential individuals.

2.2.3 Mobility

The mobility of users has great impact on the content dissemination process. In mobile social networks, users have high frequency to move around the network, which causes the available connections vary frequently. On the other side, at different locations, it is probable that users access to different content. Hence, the study on users mobility is an important component to understand the movement patterns of mobile users and further the influence on the content dissemination. The authors in [36] analyze users mobility as periodic mobility model and social network driven mobility model. As the majority of human movement, periodic mobility model describes users mobility as periodic movement between a small set of latent location. For instance, users probably stay at their working location at work time and returns home after the work. The social network driven mobility analyzes the movement behavior caused by social communications. The work in studies human's behavior in large scale disasters. In this work, the authors propose to use hidden Markov model to model the dependency between disaster behaviors and related disaster location and states.

Along with the mobility of users, the geography information can be utilized to enhance the content dissemination process as well. Geo-community is proposed in [37][38], which is based on the observation that users' interests are often highly related to geography. For instance, colleagues contact each other in the office; basketball lovers play basketball together in gyms. Hence, users with similar interests form tightly link to each other in geography related communities (i.e., geo-communities). Users thus move around several geo-communities, the mobility of which can be modeled by a time homogeneous semi-Markov model. Based on mobility model, we can derive the steady state probability of the presence of a user i in

geo-community j. According to the steady state probability of presence, we can contact user i with a degree of certainty of we wait in geo-community j for a certain long time. Therefore, the authors propose a superuser route dissemination algorithm, in which a superuser with the content moves among the geo-communities and stay in the geo-communities for a certain long time so that it can contact and directly deliver the content to users.

2.2.4 Privacy and Security

Users in MSNs may have their concern on the privacy and security during dissemination To provide the privacy-preserving data dissemination, the overlays only include the connections between nodes who trust each other. Examples of this approach are Turtle [39], Whisper [33] and MCONs[34]. Whisper [33] and MCONs[34] enable privacy preserving in group communication by limiting the communication existing between the social connections within the same group. The group membership is generated by invitation in their work.

The work in [35] proposes a privacy preserving method to improve the robustness of trust-based data dissemination. For conventional trust-based dissemination, dissemination happens only among users with certain social relationship, which is ineffective unfortunately. To bootstrap the trust-based dissemination, it is a good a idea by employing additional links between users with no social relationship connections. The additional links should provide an abstraction of privacy-preserving routing as well. The privacy preserving additional links are estimated based on anonymity and pseudonym with limited lifetime. Finally, the trusted links together with additional pseudonym links provider a random graph overlay, which can provider good performance on dissemination, privacy and reliability.

2.2.5 Incentive

The content dissemination in MSNs normally consumes a large size of physical resources from the participating users, such as, bandwidth, memory buffering, power and so on. Hence, the selfishness of the users may bring negative impact on the performance content dissemination. To overcome the impact from selfishness and other negative characters, an effective

incentive scheme is necessary in many scenarios.

Different from traditional incentive scheme in wireless ad hoc networks, the motivation in MSNs may require more concentration on the social features. Hence, users would have more incentive if they are motivated by more social benefit.

Existing incentive schemes for content dissemination in MSNs can be categorized into three categories: reputation based, tit-for-tat based and credit (victual currency) based. In reputation based scheme, dissemination services are provided to nodes depending on their reputation records. Successful dissemination behavior can increase the reputation correspondingly; vice versa. Give2Get[30] is a typical reputation based incentive scheme, which can detect misbehaving users and remove them from content dissemination. A Nash equilibria is achieved in Give2Get to prevent rational users from deviating. In Give2Get, the content details are hidden from candidate relay users before the relay users agrees to server the dissemination process. Proof of relay are also required after the selected relay users agrees to serve and receive the encrypted content.

Tit-For-Tat is a pair-wise motivation scheme, which is built on the pair wise behaviors of two users. Typically, a user will choose more generous behavior to another user if it is treated generously by the other user. Shevade et. al.[31] develop a TFT based incentive scheme for selfish users to optimize their own performance without significant degradation of network wide performance. In [31], generosity and contrition are incorporated. Generosity enables bootstrapping and absorbs transient asymmetries, while contrition prevents mistakes from causing endless retaliation. Bootstrapping happens when two users meet for the first time. Since no content have ever been successfully relayed by both users, the basic TFT prevents the start of any relay. Contrition is imported to prevent mistakes from causing endless retaliation and provide a way to return to stability after perturbation, by refraining a user from reacting to a valid retaliation to its own mistake.

Credit based incentive makes use of certain credit or virtual currency to motivate users for relaying and dissemination. The credits are normally issued and maintained by content providers or social service providers. Users successfully disseminating content would receive

certain credit as rewards. [32] proposes a novel credit based motivation scheme for advertisement dissemination. When a intended receiver (ad targeted customers) receives some copies of ads from an intermediate node, it authorizes the latter a number of virtual checks as a proof of delivery. When a intended receiver (ad targeted customers) receives some copies of ads from an intermediate node, it authorizes the latter a number of virtual checks as a proof of delivery. In addition, check can be trade to quickly cash the check with content provider.

2.3 Content Perspective

As the objects to be processed, the properties and requirement of content should be considered during the content dissemination process in mobile social networks as well. In many cases, there is no restriction on the content, which can be called free content. The free content can be received and edited by any users. And any connections can be utilized to transmit the content, regardless the bandwidth and duration provided by the connection. However, some other content has their dissemination requirements. In this section, we analyze the interesting content types that have specific properties and requirements

2.3.1 Simple Content

Simple content is the most common content type existing in content dissemination applications. Most content generated in daily social networks can be categorized as simple content, such as messages, news, pictures and so on. There is no limitation and requirement to disseminate simple content. Every user in social networks has the right to copy, edit and delete the open source content. And the simple content can be transmitted through almost any connections, regardless the connection condition, duration and distance.

To disseminate simple content in mobile social networks, the content providers need to consider more constraints from network condition and user preference, instead of content perspective. Researchers have proposed various studies on the dissemination problem for simple content. Gao et. al.[40] propose to efficiently maintain the cache freshness by organizing the caching users as a tree structure during content access. Focusing on similar

problem, Chen et. al.[41] deduce the optimal file replication strategy by further considering users' ability to meet others as a resource.

2.3.2 Streaming Content

Streaming content is also a challenging content type which has particular requirement on the connection condition during content dissemination. Different from free content, streaming content has the properties of large size, long transmission duration and high data completeness requirement. To efficiently disseminate streaming content in mobile social networks, the network should provider high speed and long duration streaming data transmission paths[42]. However, because of the wireless and mobility environment of Mobile Social Networks, many existing connections disobey the requirements. The mobile ad hoc connection quality is highly limited by distance and the power of mobile devices. In additional, user behaviors and mobility would disrupt the connection as well. Hence, to disseminate streaming content in MSNs, we need to detect the reliable dissemination paths which providers long duration and high speed transmission performance. It is an essential challenging on how to detect the reliable streaming data transmission paths in MSNs.

User preference analysis can be leveraged to predict the possible behavior acted during streaming content dissemination. By analyzing the mobility, interests on content and attitude on social influence, we can estimate the possible behavior towards the streaming content dissemination, and further detect the users who would be active during the dissemination process. The authors in [43] propose a collaborative mobile architecture to model user behaviors and stimulate user cooperation in multicast live streaming. In [44], the authors analyze user activities on live video streaming systems and identify the impact of those activities on performance. A Bayesian network is built to model user behaviors and help to enhance the live streaming system.

Studied have been taken to solve the streaming dissemination problem by optimize the caching and streaming schemes. To avoid disconnection and service breakdown caused by users' mobility in the network, Wu et. al.[45] propose a two level framework for cooperative

media streaming in MSNs. In the framework, headlight prefetching and dynamic chaining are designed to deal with the uncertainty of user movement and maximize cache utilization and streaming benefit.

Chapter 3

BAYESIAN-BASED CONTENT DISSEMINATION FRAMEWORK

In this chapter, we study the dissemination problem of large size of content such as multimedia videos.

Since videos carry abundant visual contents and information, the video content is an attractive content type deployed in plenty of applications. However, video dissemination in MSNs encounters more significant challenges in the distribution process than other types of contents such as text and picture do. First, it requires high bandwidth and reliable routing paths, which are difficult to be guaranteed in dynamic unstable mobile ad hoc networks. Second, the nature of the social networks such as user interests, connections and behaviors may have a great impact on how the video is disseminated across the physical network. For example, how one can motivate a neighbor mobile user to participant or help forwarding the video content if a direct link to the video source does not exist. Social users can be affected by many factors (e.g, battery running down) and may respond dramatically different to the same factor.

In this work, we propose a novel framework for effective video content dissemination in mobile social networks, which takes into consideration both the social features and the limited physical resources[46]. Different from other related work, our framework captures the individual user's personal decision and interaction during the process of video distribution. We analyze the factors that impact users' choices, including video content matching with users' interests, the decision or choices of other neighbor users and the physical resources (e.g., battery and bandwidth). By synthesizing those factors, we develop an effective Bayesian network model which enables each user to calculate its probability to request the video (*i.e.*, video request probability). This probability is then used to select the optimized routing path for video sharing and transmissions. To the best of our knowledge, this is the first

work in modeling the social characteristics and network resources for video sharing using the Bayesian technique.

3.1 The Framework for Video Sharing in MSN

In this section, we present our novel approaches to enhance users' experiences when sharing and receiving videos in mobile social networks. Figure 3.1 shows the proposed framework for video sharing in the Mobile Social Networks (MSN), which includes three main modules.

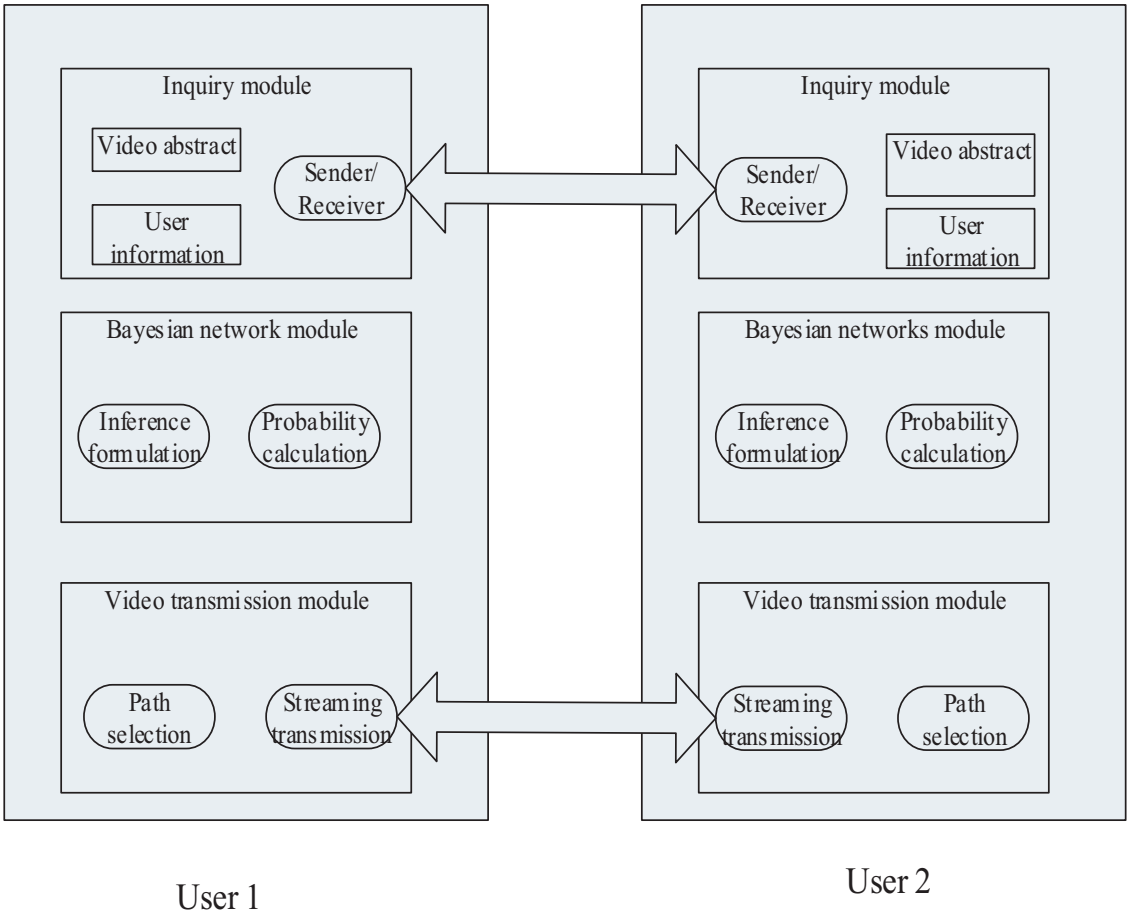


Figure 3.1. Video sharing in MSN

Inquiry module: The inquiry module is designed to inquire user information and distribute video abstracts. User information is used to find existing neighbors and formulate

the user inference in Bayesian networks. Video abstract includes a brief introduction or description of the video, video tags, video rating and file locations, which enable users to quickly read the video overview in the abstract. With this information, users can derive the video inference in the proposed Bayesian model (as shown in Section 3.1.2).

Bayesian network module: The Bayesian network module involves inference formulations and mathematical methods to calculate users' probability to request the video, denoted by video request probability.

Video transmission module: In video transmission module, we use the results from the Bayesian model to help users select effective routing paths for video transmissions.

The proposed framework requires three phases for video sharing among MSN users. First, video abstracts are distributed among the whole network. Along with this process, user information is also exchanged between neighbors (i.e., neighbor discovery). Next, the proposed Bayesian model is leveraged to help users to find the most appropriate paths to transmit videos. Finally, users establish transmission paths and the videos can be downloaded through the identified paths. To enhance the transmission performance, we also propose a multiple paths transmission scheme.

3.1.1 Abstract Distribution

In our framework, a user (or video source provider) disseminates the abstract of a video to users in the network. The distribution of the abstract is executed in a gossip fashion such that every user in the network is aware of the abstract. After receiving the abstract, each user then forwards the abstract to its neighbors who have not received the abstract. Once the abstract reaches all users in the network, the abstract distribution flow will terminate automatically.

Figure 3.2 illustrates an example of the abstract distribution process. In this example, the user U_1 is the video source provider. Neighbor users U_2, U_3, and U_4 can obtain the abstract in the first round. Users in two hops away, including U_5 to U_9 will receive the abstract in the second round. Thus the structure of the network consists of several levels

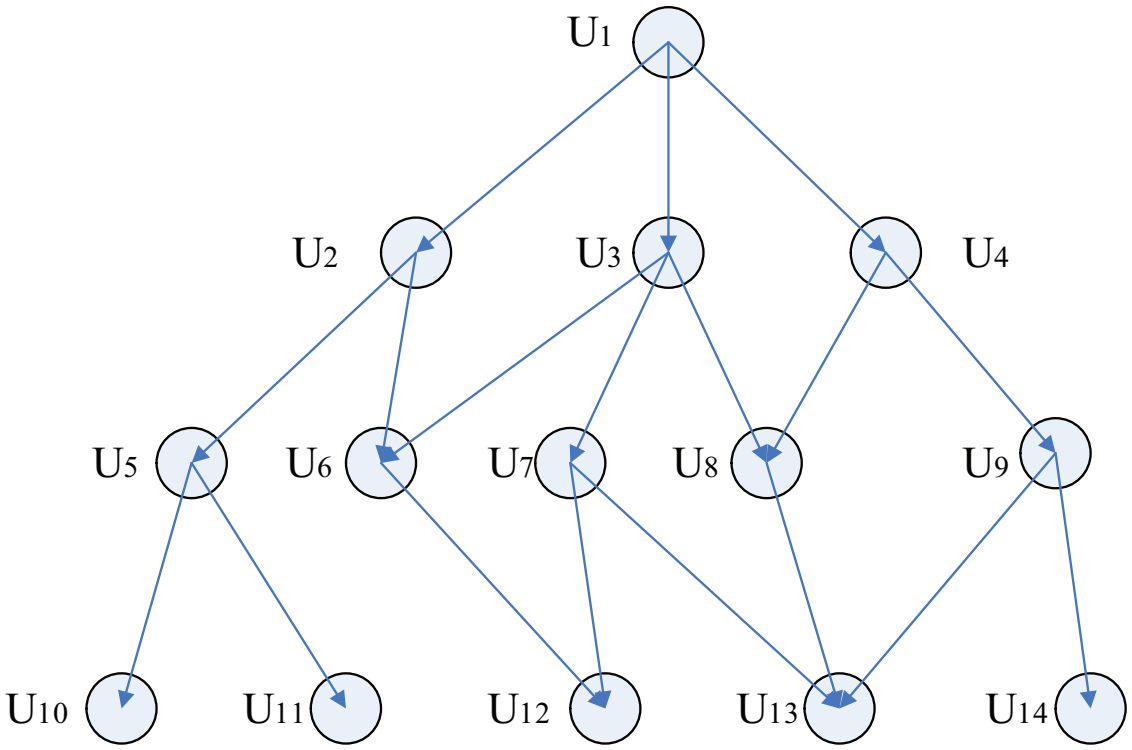


Figure 3.2. Scheme of the abstract distribution flow

according to the distance to the video source provider.

During the process of abstract distribution, when a user forwards the abstract to its neighbors, it attaches its personal information such as video request probability and resources status (e.g., battery status, bandwidth allocated to neighbors, as to be discussed in the following sections).

3.1.2 Bayesian Model

After receiving the abstract from its neighbors, a user will establish a Bayesian network model, which analyzes the influence of users' behaviors and network resource conditions. The probability that a user requests a full video version from the video source provider, namely **video request probability (VRP)**, is computed to facilitate the selection of an effective path for later video transmissions.

The video request probability calculation mainly considers three factors: confidence on neighbors, interest match, and the resource status.

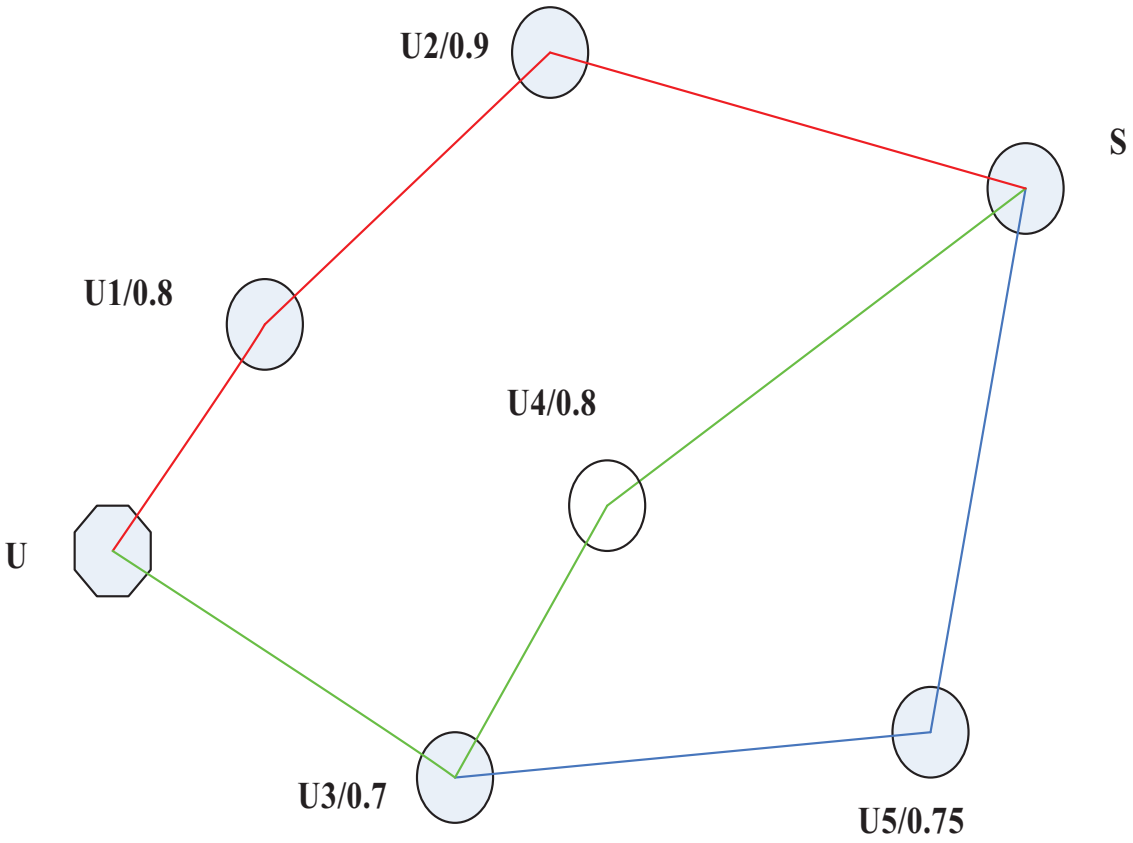


Figure 3.3. Example for multi-path transmission

(i) The **confidence** on neighbors represents how a user trust its neighbors and the neighbors' choices. A larger confidence indicates a higher probability in trusting the neighbors and thus making similar choice as its neighbors.

(ii) The **match** between the user's interest and the video content shows how likely the video will draw the user's attention. A higher level of match indicates a larger probability the user requests downloading the video.

(iii) The resource **status** of mobile devices, includes the battery level and available bandwidth. Abundant resources would encourage the user to request and watch more videos. With fully charged battery, naturally, a user would likely spare more resources for the video transmission. Otherwise, the resources allocated for the video transmission may be limited.

We use $U = Y$ to represent the event that user U requests the video and contributes to the video forwarding/transmission, and $U = N$ to denote user U rejecting the video. Hence

$P(U = Y)$ is defined as the video request probability. $\Psi_U = \{U_1, U_2, ...U_n\}$ is the neighbor set of user U. In addition, we use I to denote the match of the user's interest with the video content. S is used to indicate the resources status of the user's mobile device. In our Bayesian model, both I and S are variables.

According to the above definitions, we can calculate the video request probability $P(U = Y)$ as:

$$P(U = Y) = \sum_{\psi_U, I, S} P(U = Y|\psi_U, I, S) \cdot P(\psi_U, I, S) \tag{3.1}$$

where $P(U = Y|\Psi_U, I, S)$ is the conditional probability that user U requests a video and contributes to the video forwarding/transmission. Since a user may have different interests from its neighbours and the interests generally have no relation with the physical resources status of the mobile device, these three factors can be regarded independent from each other. Accordingly, the conditional probability distribution can be rationally derived as:

$$P(U = Y|\Psi_U, I, S) = \alpha P(U = Y|\Psi_U) + \beta P(U = Y|I) + \sigma P(U = Y|S) \tag{3.2}$$

where $\alpha + \beta + \sigma = 1$ and $0 < \alpha, \beta, \sigma < 1$

α, β, σ are three coefficients indicating users' preference towards those three factors in mobile social networks. A bigger α implies that users in this mobile social network emphasize more on user interactions (or confidence). A mobile social network with a bigger β pays more attention to video content matching. The video has higher probability to be requested by those who are really interested in it. Similarly, a bigger σ indicates a larger impact from the physical resources status of mobile devices.

The conditional probability $P(U = Y|\Psi_U)$ in Eq. (3.2) can be calculated based on the set of conditional probability of $P(U = Y|U_i = u_i)$, where U_i is the ith neighbour of user U,

and u_i is a possible decision (or choice) [1] of U_i (i.e., Y or N):

$$P(U = Y|\Psi_U) = \sum_{U_i \in \Psi_U} f_{U_iU} P(U = Y|U_i = u_i) \tag{3.3}$$

In the Equation (3.3), f_{U_iU} is the weight of the social relationship between user U_i and user U. To quantize the weight of the social relationship, we investigate the contact/communication history among users. It is naturally supposed that people in closer relationship will contact with each other more frequently. We record the number of conversations occurring between users and normalize them to calculate the weight.

$$f_{U_iU} = \frac{N_{U_iU}}{\sum_{U_j \in \Psi_U} N_{U_jU}} \tag{3.4}$$

In Eq. (3.4), $N(U_iU)$ is the number of the conversations in history between user U_i and U.

The other factors needed to derive the conditional probability $P(U = Y|\Psi_U, I, S)$ are $P(U = Y|I)$ and $P(U = Y|S)$. The conditional probability $P(U = Y|I)$ means how user's decision is influenced by the match between the user's interests and the content of the video described in the video abstract. Similarly, $P(U = Y|S)$ indicates the impact of physical resources such as battery and bandwidth on the user's decision to request the video. In the following, we show how to calculate the video request probability while practically formulating these conditional probabilities in our framework.

Calculating Video Request Probability Based on the analysis above, we can derive the conditional probability $P(U = Y|\Psi_U, I, S)$ as:

$$\begin{aligned} P(U = Y|\Psi_U, I, S) \quad = \quad & \\ & \alpha \sum_{U_i \in \Psi_U} f_{U_iU} P(U = Y|U_i) + \beta P(U = Y|I) + \sigma P(U = Y|S) \end{aligned} \tag{3.5}$$

[1] The terms decision and choice are used interchangeably throughout the paper.

A user U will receive the video request probabilities of its neighbors, i.e., $P(U_i = Y)$, from each neighbor U_i. Suppose ψ_U is the set of possible choice variables of the neighbors in Ψ_U, which is $\{Y, N\}$ in this work. Let I', S' denote the possible value of interest match I and resources status S, respectively.

Considering that $P(\psi_U)$, $P(I)$, $P(S)$ are independent, and different users are also independent with each other, we can get:

$$\begin{aligned} P(\psi_U, I, S) &= P(\psi_U) \cdot P(I) \cdot P(S) \\ &= \prod_{U_i \in \Psi_U} P(U_i) \cdot P(I) \cdot P(S) \end{aligned} \tag{3.6}$$

Thus, the formulation of video request probability in Equation (3.1) can be derived as:

$$\begin{aligned} P(U = Y) = & \\ & \alpha \sum_{\Psi_U, \psi_U} f_{U_i U} P(U = Y | U_i) \prod_{U_k \in \Psi_U} P(U_k = u_k) \\ & \cdot \sum_{I,S} P(I = I') \cdot P(S = S') \\ & + \beta \sum_{I} P(U = Y | I = I') P(I = I') \cdot \sum_{\Psi_U, S} \prod_{U_i} P(U_i) P(S = S') \\ & + \sigma \sum_{S} P(U = Y | S') P(S') \cdot \sum_{\Psi_U, I} \prod_{U_i} P(U_i) P(I = I') \end{aligned} \tag{3.7}$$

where,

$$\begin{aligned} & \sum_{\Psi_U, \psi_U} f_{U_i U} P(U = Y | U_i) \prod_{U_k \in \Psi_U} P(U_k) \\ &= \sum_{\Psi_U} f_{U_i U} P(U = Y | U_i) P(U_i) \cdot \sum_{\psi_U} \prod_{\Psi_U - \{U_i\}} P(U_k = u_k) \\ &= \sum_{U_i \in \Psi_U} \sum_{u_i} f_{U_i U} P(U = Y | U_i = u_i) P(U_i = u_i) \end{aligned} \tag{3.8}$$

In Equation (3.8), $\Psi_U - \{U_i\}$ is the neighbor set of user U except U_i.

For a particular user, the interest is fixed and known. Given a video, the interest match

of the user can be certainly calculated, which is denoted by $I(U)$. Therefore, only the variable I with the same value of $I(U)$ is available. Hence, the probability $P(I)$ can be represented as:

$$P(I = I') = \begin{cases} 1: & \text{if } I' = I(U) \\ 0: & otherwise \end{cases}$$

In this work, several tags (or keywords) are created in video abstract to represent the video content. Similarly, a user will initialize its interests, represented by several tags. Therefore, we can match user interest and video content by calculating the common tags between the user's interests and video content, and calculate $I(U)$ as Equation (3.9),

$$I(U) = \frac{No(T_I \cap T_C)}{No(T_I)} \tag{3.9}$$

where $No(T_I \cap T_C)$ is the number of common tags between the user's interests and the video content, and $No(T_I)$ is the total number of tags the user is interested in.

Similarly, we can check the hardware resource information and obtain the physical status of the user's mobile device, as $S(U)$. Then the probability $P(S)$ can be described as the following formula.

$$P(S = S') = \begin{cases} 1: & \text{if } S' = S(U) \\ 0: & otherwise \end{cases}$$

Therefore, the factors involving interests and resources status can be described in Equation (3.10) and Equation (3.11), respectively:

$$\sum_{I} P(U = Y|I)P(I) = P(U = Y|I(U)) \tag{3.10}$$

where $I(U)$ is the interest matching between user U and the particular video.

$$\sum_{S} P(U = Y|S)P(S) = P(U = Y|S(U)) \tag{3.11}$$

where $S(U)$ is the current resources status of user U.

Accordingly, Equation (3.7) can be expressed as,

$$P(U = Y) = \sum_{U_i \in \Psi_U} \sum_{u_j} \alpha \cdot f_{U_i U} P(U = Y | U_i = u_j) P(U_i = u_j) + \beta P(U = Y | I(U)) + \sigma P(U = Y | S(U)) \tag{3.12}$$

Inference Learning The Inference formulation module provides a mechanism for a user to formulate the conditional probabilities used in the Bayesian network. Three conditional probabilities are required in the learning module: $P(U|U_i)$, $P(U|I)$ and $P(U|S)$.

The conditional probability $P(U|U_i)$ indicates the impact from the decision of its neighbor U_i. In specific, $p(U = Y|U_i = Y)$ implies the probability that user U requests the video if neighbor U_i requests the video, which depends on U_i's attitude on others' choices and the bandwidth b_{UU_i} contributed to user U by user U_i. The personal preference is a parameter independently formed by the user himself. For a given user U, $p(U = Y|U_i = Y)$ is directly proportional to b_{UU_i}, while $p(U = N|U_i = Y)$ is inversely proportional to b_{UU_i}.

$P(U|I)$ indicates the influence of interest match on users' decision. Generally, a higher interest match leads to a higher video request probability. $P(U|I)$ can be formulated as a linear function of I. The derivation of $P(U|S)$ is similar to that of $P(U|I)$, which shows the impacts of the resources status of mobile devices on users' decision.

When a new user U joins the system, it first chooses its preference, including the parameters α, β, σ and interests. The $P(U|U_i)$ is also formulated by the user. The friend relationship f_{U_i} are also initialized. After receiving the video information and video request probability $P(U_i)$ of its neighbor U_i in the abstract dissemination process, the new user can calculate the video request probability $P(U)$ based on Equation (3.12).

During the transmission phase in Sec. 3.1.3 and Sec. 3.1.4, several parameters are updated: f_{ij}, $P(U_i = Y)$, and S. After each successful transmission among two users, the weight of relationship between them is updated as in Equation (3.4).

3.1.3 Video Request and Transmission

When a user sends out a video request, our framework provides an efficient approach to help the user select the most effective transmission paths based on the video request probability.

First, user U selects the neighbor with the highest video request probability, say user N, and sends the download request to user N. When neighbor N receives a video download request, it will check whether it has already obtained a copy of the video. If yes, neighbor N transmits the video data backward to user U. Then the request forwarding process is terminated. Otherwise, user N makes decision to forward the request depending on the probability $P(N = Y)$. User N will forward the request to N's neighbors only when user N reaches a decision that it also wants a copy of the video. If user N decides not to participate the video sharing process, the request forwarding process is terminated. A termination signal is sent back to the previous neighbor. When a user receives a termination signal, it will choose the neighbor with the next highest video request probability to resend the download request. The process will repeat until a download request arrives at the video source provider and a successful transmission path is established. Clearly, if the original video request user U receives termination signal from all U's neighbors, user U will not find any transmission path for video request.

When a successful transmission path is established, streaming video transmission starts. The video is divided into several streaming segments. Each segment is transmitted and played, as the protocols in VOD streaming or the streaming standards like H.264 and MPEG-4.

3.1.4 Multiple Streaming Transmission

In mobile social networks, mobile devices are usually not able to own stable and high download/upload bandwidth. Hence, it is necessary to explore better Quality of Service (QoS) according to different network conditions. To achieve this, we propose Multiple Streaming Rate (MSR) or Multiple Description Coding (MDC) to provide a smooth and

effective streaming in MSN. MSR and MDC allow the video source to construct several independent descriptions of the same video as discussed in [47] [48].

We design a multi-path transmission scheme based on the MDC technique. After the video abstract distribution process, each user U holds a list L_U in which the transmission probabilities of its neighbors are sorted as Equation (3.13)

$$L_U = \{P(U_i = Y)\}; U_i \in \Psi_U \tag{3.13}$$

We then use this list to select appropriate users and paths for multi-path video transmissions. In specific, the path with the highest transmission probability is selected to transmit the basic description of the video. Then, additional one or multiple paths are chosen to transmit the enhanced video descriptions. The basic idea is to select the users with higher transmission probabilities in Equation (3.12) to from the additional paths.

Figure 3.3 demonstrates an example of this process, where User U is the video requestor, and S is the video source provider. U_1-U_5 are the intermediate users in the network. The decimal numbers in the figure are the corresponding video request probability. At first, user U attempts to select {U, U_1, U_2, S} as the primary transmission path. If this path is successfully established, user U will try to build a second path through U_3-U_5. According to the sequence in L_{U_3}, U_4 or U_5 will be selected on the second path. If both U_4 and U_5 or U_3 refuse the request, no secondary path would be established.

3.2 Simulation and Performance Analysis

In this section we test the Bayesian models and related parameters in two simulation platforms: numerical analysis and OPNET simulation [49]. We first present the result from the numerical analysis and OPNET simulation [49] in a tree or grid network. Then we evaluate the performance of the proposed schemes in networks with a random network topology.

In the simulation, each user randomly chooses its confidence on neighbors' choices

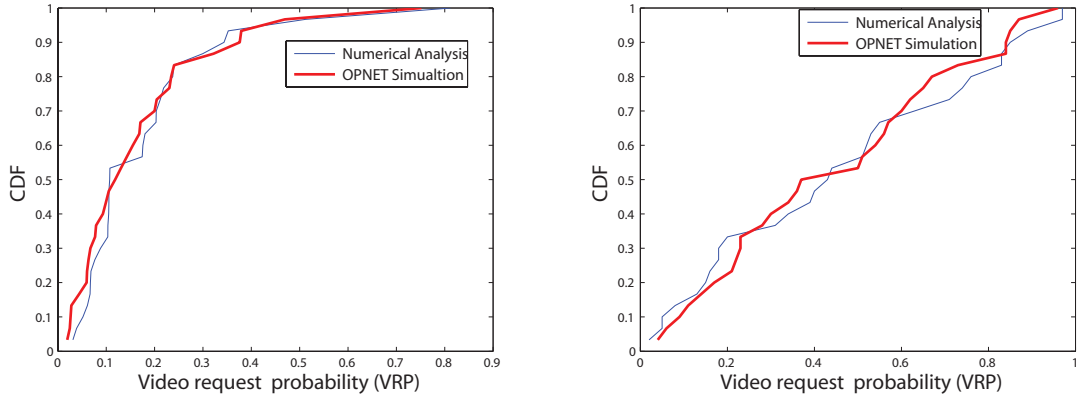
$P(U_j|U_i)$, interest on the video $I(U)$ and resource status $S(U)$. We can vary the three coefficients in Equation (3.13): α, β and σ, according to different network scenarios. Table 3.1 shows four typical parameter settings we test in the experiments. Clearly, in Case 1, users are only affected by neighbors' choices and users have no content preferences and no concern on physical resources. In Case 2, video interest and physical resource status are the only factors influencing users' choices. The users in Case 3 evenly treat the three factors. In Case 4, users care more about social confidence than video interest and physical resources. Note that the factors of interest match and resource status are considered evenly as both are independently and randomly generated by each user.

Table 3.1. Parameter setting used in simulation

Case	α	β	σ	Characteristics of the network
1	1	0	0	Only the factor of social confidence (i.e., confidence on neighbors) is considered
2	0	0.5	0.5	The social confidence factor is ignored while the interest match and resource status are equally considered
3	0.33	0.33	0.33	The three factors are equally considered
4	0.5	0.25	0.25	The social confidence is the primary factor while the interest match and resource status are equally considered

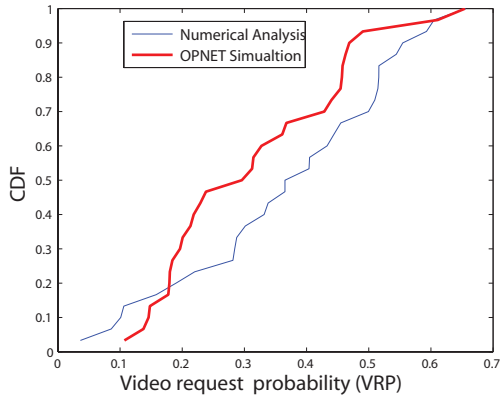
3.2.1 Performance Evaluation in Tree and Grid Networks

To evaluate the performance of the proposed schemes, we implement the Bayesian model in networks with a tree or grid topology. The tree network is a binary tree network with 31 users and the root of the tree is the video source provider. The grid network consists of 25 users forming a 5×5 grid. The center user in this grid network is the video source provider. For both networks, we draw the Cumulative Distribution Function (CDF) from the numerical analysis (i.e., Equation (3.5-3.13)) and OPNET simulation to study the distribution of users' video request probability. In specific, we use x-axis to represent the potential values of video request probability (VRP). The y-axis is the CDF of the video request probability, which denotes the percentage of users in a network having a video request probability less or equal

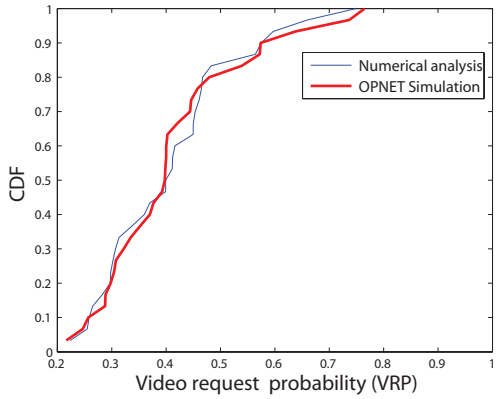


(a) Case 1 ($\alpha = 1$, $\beta = \sigma = 0$)

(b) Case 2 ($\alpha = 0$, $\beta = \sigma = 0.5$)



(c) Case 3 ($\alpha = \beta = \sigma = 0.33$)

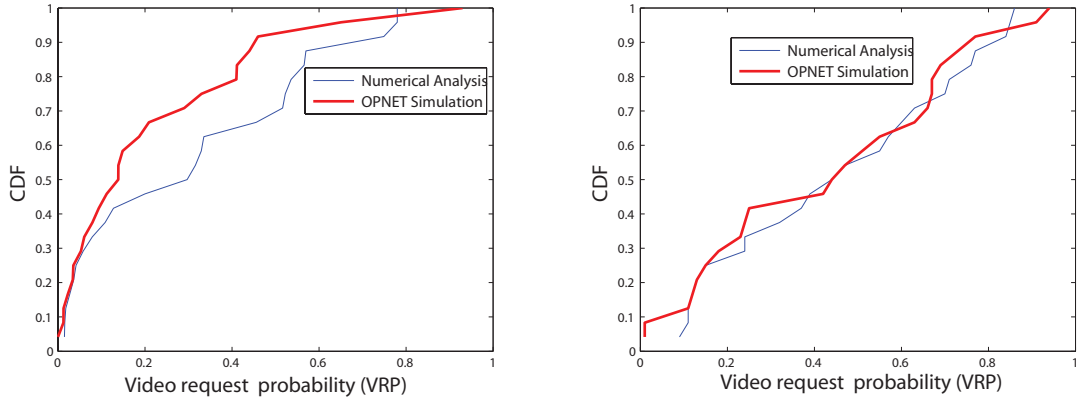


(d) Case 4 ($\alpha = 0.5$, $\beta = \sigma = 0.25$)

Figure 3.4. Video request probability (VRP) distribution in a tree network

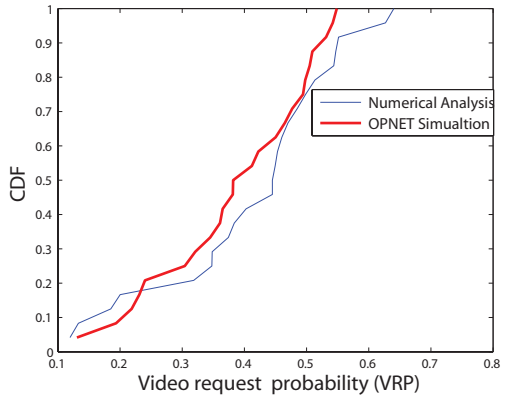
to a particular value.

Figure 3.4 and Figure 3.5 show the results of video request probability distribution in the tree and grid network, respectively. For Case 1 in the tree network, Figure 3.4(a) demonstrates that the CDF curve from OPNET simulation matches well with the results from the numerical analysis. Both curves are close to a logarithmic curve. In the binary tree network with Case 1 ($\alpha = 1$), the video request probability is solely determined by the confidence on neighbors. In other words, a user in the tree network is influenced by only the choice (i.e., VRP) of its parent, when the video request probability is calculated. For a user U whose distance to the root is d, its decision is influenced by d users along the path from the root to user U. The bigger d is the less opportunity U will request the video due to the

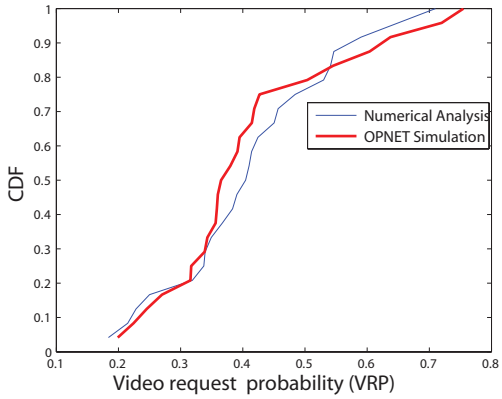


(a) Case 1 ($\alpha = 1$, $\beta = \sigma = 0$)

(b) Case 2 ($\alpha = 0$, $\beta = \sigma = 0.5$)



(c) Case 3 ($\alpha = \beta = \sigma = 0.33$)



(d) Case 4 ($\alpha = 0.5$, $\beta = \sigma = 0.25$)

Figure 3.5. Video request probability (VRP) distribution in a grid network

increasing accumulation of negative opinions. When $\alpha = 1$, Equation (3.13) is converted to:

$$P(U = Y) = \sum_{u_j} P(U = Y|U_i = u_j)P(U_i = u_j) \tag{3.14}$$

where U_i is user U's parent node. Suppose the average of $P(U = Y|U_i = u_j)$ is ϵ, then the children of user U_i including U have similar VRP values as $\sum_{u_j} \epsilon_{u_j} P(U_i = u_j)$. Likewise the children of user U have similar VRP values as $\sum_{u_j} \epsilon_{u_j} P(U = u_j)$. Recursively, the VRP value of a user is closely exponential to the distance to the root (with a base less than 1). Since the number of users with a distance d to the root is exponential to d in the tree network, the CDF function of the VRP values shows a logarithmic curve as in Figure

3.4(a). Figure 3.5(a) shows the CDF of video request probability in the grid network with Case 1. The user in the grid network is influenced by more neighbors (mostly two). Each neighbor imposes different influence on the user, which makes the curve in the grid network less close to a logarithmic curve.

The results in a tree network and grid network with Case 2 are presented by Figure 3.4(b) and Figure 3.5(b), respectively. In both figures we can see that the CDF curve is linear to the video request probability, indicating the VRP value is uniformly distributed. This is because $\alpha = 0$ makes VRP calculation totally ignoring the factor of neighbor confidence (as well as the network structure). In fact, each user only cares about the interest match and resource status, which are randomly generated by each user. Therefore, users' decisions are not affected by other users, and the CDF curve is uniformly distributed in both tree and grid networks.

Figure 3.4(c) shows the scenarios that users evenly care about the three factors as the Case 3 in Table 3.1. Influenced by both users' confidence and personal interest/status, the CDF curve lies between the logarithmic curve and linear curve in the tree network. Figure 3.5(c) shows a similar result in the grid network. The similar conclusion can also be observed from the results on Case 4 in Figure 3.4(d) and Figure 3.5(d), where all three factors are considered. Since users in Case 4 put more weight (i.e., bigger α value) on the social confidence factor, the curves in Figure 3.4(d) are more close to a logarithmic curve than the curves in Figure 3.4(c).

3.2.2 Performance Evaluation in Random Networks

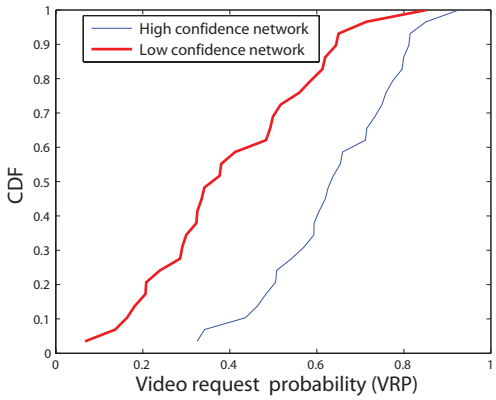
In a random network, the numerical analysis is intractable and hence we implement the framework with OPNET. There are 30 users in the random network and the average node degree, denoted by Δ, can be 5 or 2.4. We call the network with $\Delta = 5$ as a densely connected network and the network with $\Delta = 2.4$ as a sparsely connected network. By comparing the performance of those networks, we can study the influence of network connections on users behavior.

In the OPNET implementation, the user's confidence ($P(U|U_i)$) is randomly generated to simulate two typical community: close-knit MSN and loose-knit MSN [50]. We call the close-knit MSN as the high confidence network where users trust neighbors' choices more and more likely prefer the video accepted by their neighbors. On the other hand, the users in a loose-knit MSN, called as low confidence network, will be less influenced by the decision of their neighbors.

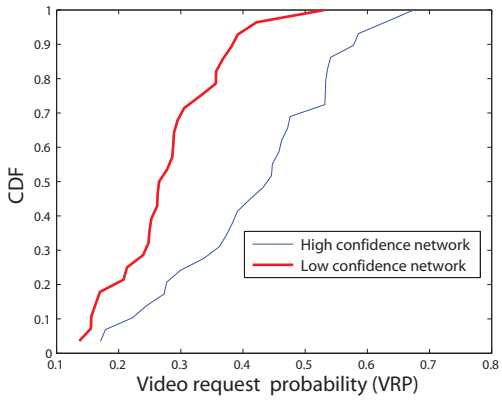
Meanwhile, we also experiment different Bayesian parameters settings as shown in Table 3.1. As explained above, Case 2 is independent of the network structure. Hence, Case 2 in the random network yields similar results as the tree/grid network. which is omitted in this section. From hereafter, if not other specified, the results are based on Case 4 in a densely connected and high confidence random network.

Impact of Bayesian Parameters Figure 3.6 shows the results for different MSN types, where x-axis represents the video request probabilities and y-axis is the cumulative distribution function (CDF) of the proportion of users with the video request probabilities. We can see the users in MSN with high confidence have higher overall video request probability than users in MSN with low confidence. Figure 3.6(a) shows that the average probability is around 0.6 in the high confidence network and 0.4 in the low confidence network, which implies users in high confidence networks likely benefit from the high video request probability and have better success rate to obtain interested videos. This can be explained as follows. In a high confidence network, a user closer to the video source has larger impacts on both close-by neighbors and far-away users, resulting in more users with similar video request probability. Among the three cases in Figure 3.6, Case 1 yields the highest average video request probability and Case 3 has the lowest average VPR. Therefor, a user may have a higher probability to request the video if the user relies more on neighbors' recommendation.

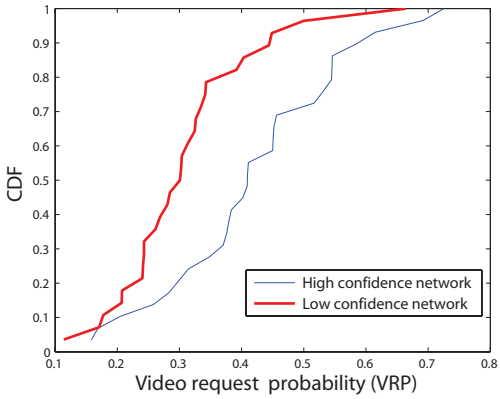
Impact of network node degree Figure 3.7 shows the performance comparison between the densely and sparsely connected networks. Clearly, the VPR in a densely connected network significantly outperforms the sparsely connected network for Case 1 or Case



(a) Case 1 ($\alpha = 1$, $\beta = \sigma = 0$)



(b) Case 3 ($\alpha = \beta = \sigma = 0.33$)

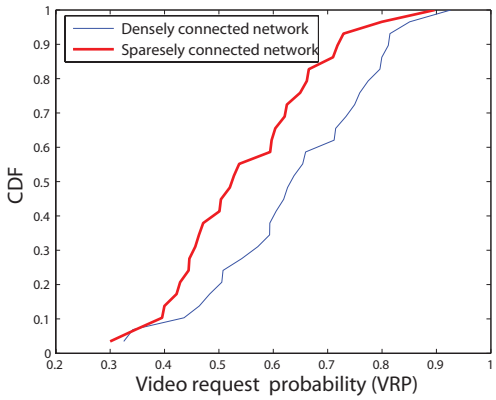


(c) Case 4 ($\alpha = 0.5$, $\beta = \sigma = 0.25$)

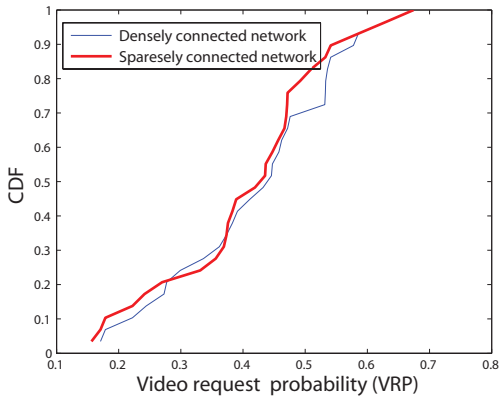
Figure 3.6. Video request probability distribution in a random network

4. However, this is not true for Case 3. This is because the impact of node degree can be more significant only if the user cares more about the neighbors' choices (i.e., larger α). We also find that the video request probabilities vary more in sparsely connected networks due to the fact that every neighbor in sparsely connected networks can impose relatively a higher influence on users decision than that in densely connected networks.

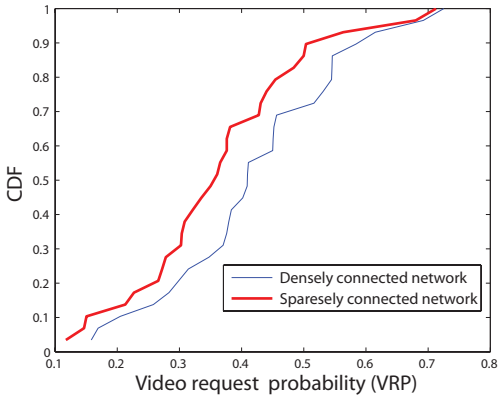
Single-path and Multi-path Transmission To investigate the transmission performance of our framework, we randomly pick a user as the video source provider and pick 10 users to request the video. Figure 3.8 shows the average transmission time versus the hop numbers to the source provider, when the single-path scheme is employed. From the figure



(a) Case 1 ($\alpha = 1$, $\beta = \sigma = 0$)



(b) Case 3 ($\alpha = \beta = \sigma = 0.33$)



(c) Case 4 ($\alpha = 0.5$, $\beta = \sigma = 0.25$)

Figure 3.7. Video request probability distribution in random networks with different node degree

we can see that the transmission time used is about $250s - 280s$, which is less than the video play length (i.e., 10 minutes). This implies that the single-path scheme is efficiently and feasible for real applications. In addition, the transmission time is almost linear to the hop distance, which indicates the transmission delay and the processing delay in mobile devices are the major delay in our system.

The results of the multi-path transmission scheme is shown in Figure 3.9, where x-axis is the IDs of the user selected and is sorted according to the number of established paths. User 1 and 2 only establish a single transmission path. Two transmission paths are built by user 3-7. User 8 and 9 can successfully build 3 transmission paths and user 10 has video

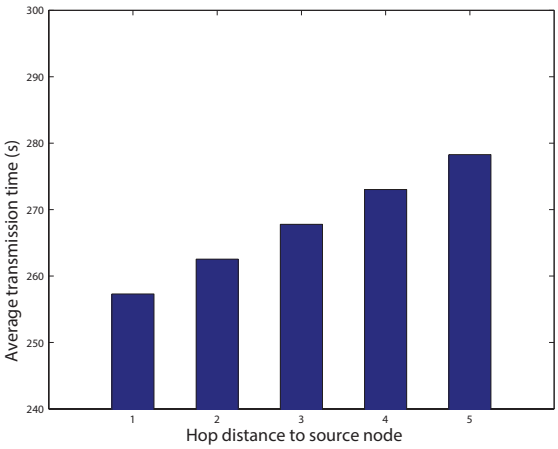


Figure 3.8. Single-path transmission

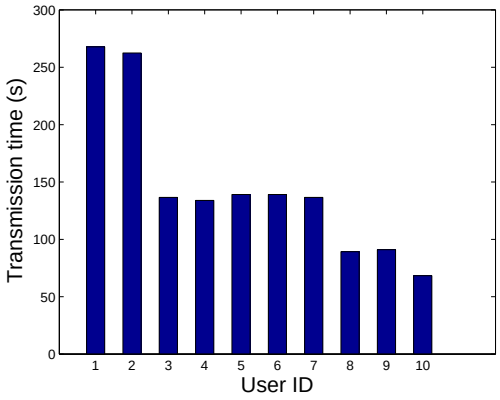


Figure 3.9. Multi-path transmission

transmission 4 paths concurrently. As we can see from the simulation, a half of the 10 nodes can establish two concurrent transmission paths and 4 or more transmission paths are rare. Multiple concurrent transmission paths can help reduce the transmission time. However the reduction in transmission time is not proportional to the number of concurrent paths due to the possible bandwidth competition among the concurrent paths.

Chapter 4

SEMI-CONTROLLED CONTENT DISSEMINATION

The content disseminated in mobile social networks has different types of generators and receivers, transmission pattern, expectation from generators and so on. Hence, there may be specific constraints or limitation on the dissemination process. Some content types such as news and blogs tend to draw public interests, and the content holder/receiver has a full control of the content and can re-disseminate the content freely. For some other types of content, the content creators may want to limit or control the dissemination process due to the value, cost or resources associated with the content creation. For example, a merchant may distribute a certain number of coupon brochures to potential customers; a bookseller delivers some free books to the book fans; a conference organizer may send a limited number of invitations to potential interested people. When received the information, a user will adopt the information which brings some profit/reward to the content provider. For instance, after receiving a coupon, a user will drop by the local store and purchase goods, which increases the income of the coupon provider. We call such process of information dissemination as Semi-controlled Authorized Information Dissemination (SAID). A common feature of such content dissemination is that the content has certain copyright protection, and can only be generated by the content creators or provider (CP). The CP will control the number of information copies to be disseminated in the networks, and disable the copy capability of the content receivers.

Since there are a limited number of authorized content copies generated by the content creators or providers in SAID, the content providers have much incentive to deliver those content copies to the users interested in or highly related to the content, who potentially bring more profit/reward to the content provider than others. For example, a brand loyal customers has higher probability to purchase more goods than other non-loyal customers. For this SAID

problem, current conventional dissemination approaches such as online publishing/applying fails due to the following reasons. First, the interested users may not be aware of the content published by the CP. Second, the content might be applied by the greedy uninterested users who may not adopt the content. Last but not least, the CP's management and maintainess cost is increased. Hence, the content providers may prefer to leverage the social network users to help disseminate and forward the content to those interested users who may or may not have direct connections to the CP.

In this work, we investigate the Semi-controlled Authorized Information Dissemination (SAID) in Content-based Social Networks. There are several characteristics in the semi-controlled authorized information dissemination. First, the total number of information copies existing in the network (e.g. the number of conference invitations to deliver), is limited and fixed. Second, each user u has a different level of personal interests (denoted by weight: w_u) in the content and may retain one or more content quotas (denoted by quota: t_u) if it receives the information. When receiving a number of content copies, each user u will retain t_u content copies, and forward the rest to others. Third, for a user (say D in Figure 4.1) to receive a copy of the content, there must exist some social users (e.g., A, E, in Figure 4.1) who would like to cooperate and disseminate the content copy to the user (*i.e.*, D) when receiving the content. In other words, there must be a *dissemination flow* from the content provider to the user. Considering these constraints, the CP must explore the best choice of the content receivers to maximize the total weights of the content receivers when disseminating the authorized information in content-based social networks.

We can use Figure 4.1 as an example to illustrate the process of authorized information dissemination. As shown in Figure 4.1, there is a content provider (CP) who would like to disseminate a limited number of (say 6) content copies to interested users. Each user has a different level of personal interests in the content (i.e., *weight* w), and may hold different number of quotas or copies according to its own need (i.e., *quota* t). For instance, user A is very interested in the content at level $w_A = 4$, and will retain $t_A = 2$ copies if receiving the content. Obviously, the goal of the content provider is to deliver the 6 content copies to the

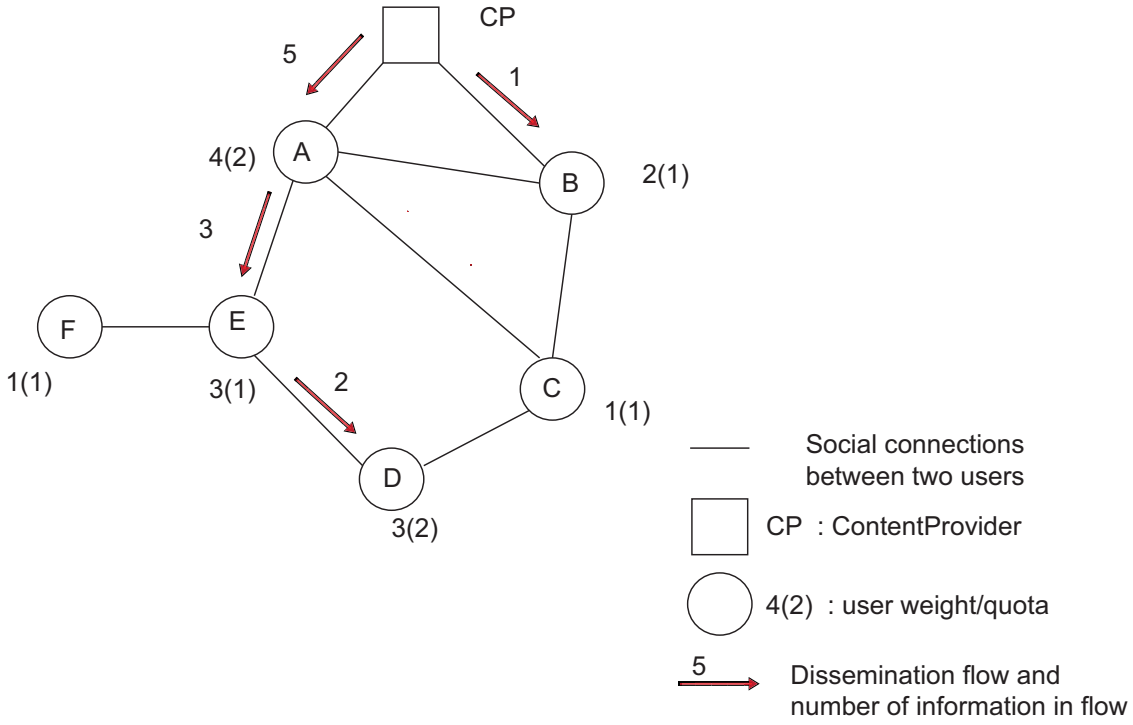


Figure 4.1. An example of SAID flow

users with high interests in the content, which can be measured as the total weight of all the users received a copy of the content. Therefore, the best strategy for the content provider is to disseminate the 6 content copies to users A, B, D, E as shown by the dissemination flow in Figure 4.1. Along the dissemination flow, each content receiver has a connected path to the CP to receive one or more copies of the authorized content from the CP. In specific, user A will receive 2 content copies directly from the CP with a weight of $w_A = 4$, Similarly user B receives one copy with a weight of $w_B = 2$. User E receives one copy via user A from the CP and contribute a weight of $w_C = 3$. With a weight of $w_D = 3$, user D will receive 2 copies via user E and A from the CP.

4.1 Maximum Weighted Connected subgraph with node Quota (MWCQ)

In this section, we first describe our methodology to represent the content and user interests, by employing the approaches used in Content-Based Networks. The weight of user nodes can then be calculated by measuring the match between the information content and

user interests. Then we formally define the Maximum Weighted Connected subgraph with node Quota (MWCQ) problem.

4.1.1 Weight Calculation

Each information or content has properties on multiple topics such as types, categories, locations, etc. These information properties are used to compare with users' interests and a better match indicates the corresponding information is more attractive to the users. To effectively measure the match between information properties and user interests, we take advantage of the methodology of the naming scheme in Content-Based Networks(CBN)[51][52][53]. Specifically, the properties of information are defined and named within a naming space. The naming space has a hieratical structure. For example, an information instance may have the property on category topic as *"category/goods/electronic device/laptop/apple/macbook air"*. In this case, the naming space has a 6-level structure in "category". Similarly, users' interests are named within the same naming space.

We use the largest matching level shared by user's interest and information properties to measure the match. Hence, if a node has interest in category topic as *"category/goods/electronic device/laptop/lenovo/Thinkpad x220"*, we will have a matching level 4 on the topic "category" between this user and the previous information instance. A higher level matching between them indicates a higher confidence that a user is interested in the information, thus a higher weight of the user on the information can be assigned.

Suppose the function $f_k(I, U)$ returns the match level of topic k between the interest of user node U and information I. The weight of user node U on information I is calculated as:

$$w_U(I) = \sum_k f_k(I, U) \tag{4.1}$$

where k is the index of information topic. If a user node has a higher weight, there is a higher probability that the node brings high profit/reward to the content provider.

4.1.2 MWCQ Problem Formulations

Given a social network G with N users, there exists a source node s to generate and disseminate R copies of authorized content. Each user u has a weight ω_u $(1 \leq u \leq N)$ calculated by matching the properties of information content and the social interests of the corresponding user. Assume that quota t_u is the number of content copies node u will retain. A matrix $C_{N\times N}$ represents the connectivity among the users in the network.

$$C_{uv} = \begin{cases} 1 \;\; ; & \text{if } u \neq v,\ u,v \text{ are connected} \\ 0 \;\; ; & \text{otherwise} \end{cases}$$

The objective of the problem is to achieve the maximum weight sum of the users who receives the content. The objective can be formulated as Equation (4.2), where δ_u is used to indicate whether the node u receives the information or not. $\delta_u = 1$ means that user node u receives the information and retains t_u information copies. On the other hand, $\delta_u = 0$ indicates user node u does not participate the information dissemination process at all (i.e., does not receive the information and is not on the path of dissemination flow). Suppose x_{uv} is the number of information copies flow from node u to node v. Then, $\sum_{v\neq u} x_{uv}$ is the number of information copies node u receives, and $\sum_{v\neq u} x_{uv}$ is the number of information copies node u sends out. Accordingly, we can formulate the Maximum Weighted Connected subgraph with node Quota (MWCQ) problem as follows.

$$\max \sum_{u} w_u \delta_u \tag{4.2}$$

Subject to

$$x_{uv}(C_{uv} - 1) = 0 \tag{4.3}$$

$$\sum_{v\neq u} x_{vu} - \sum_{v\neq u} x_{uv} = \delta_u \times t_u, \forall u, v \tag{4.4}$$

$$\delta_u \leq \sum_{v\neq u} x_{vu} \leq R\delta_u \tag{4.5}$$

$$\sum_{v \neq s} x_{sv} = R \tag{4.6}$$

$$x_{uv} \in \mathbb{N}, \delta_u \in \{0, 1\} \tag{4.7}$$

The constraint in Equation (4.3) means that the dissemination flow only exists between nodes with social connections. The node quota, i.e., the number of copies retained by a node, is constrained by Equation (4.4). The constraint in Equation (4.5) represents that one node will retain information copies only when there is a dissemination flow reaching the node. In other words, if a node e receives the information, there must be a path from the source to node u and all the intermediate nodes also retain some copies of the information. As shown in Figure 4.1, node D receives the information if and only if there exists a path to CP (i.e., $CP \rightarrow A \rightarrow E \rightarrow D$) and intermediate nodes (i.e., A, E) also retain copies of the information. Finally, the number of information copies in the network is R as shown in Equation (4.6).

Theorem 1. *The MWCQ problem is NP-complete.*

In the following, we sketch the proof of Theorem 1 by converting the NP-complete Steiner tree problem to the MWCQ problem.

The Steiner tree problem is defined as follows: given a connected undirected graph $G = (V, E)$, a subset $S \subset V$, and a weight set for each edge in E; find a connected subgraph $G' = (V', E')$ with minimum sum of the edge weights, where $S \subset V'$. From the graph G, we can construct the graph G_M by:

(a) For any edge e with weight w_e between node u and v, we add a new node v_e with weight $W - w_e$ and replace edge e with new non-weighted edges between u and v_e as well as v_e and v. W is a value larger than any edge weight in G.

(b) Add weight $2|V| \cdot W$ to all nodes in S.

(c) Add weight W_B to all nodes in S; $W_B \gg W$.

(d) Set the quota of each node as 1.

As a result, we form a dual graph G_M of the graph G in Steiner tree problem. Then we set the source node as one node v in S, and solve the MWCQ problem by setting information

copies as v, where $2|S|-1 \leq v \leq 2|V|-1$. The optimal result of Steiner tree problem would be the largest result of the set of MWCQ problems by shuffling j between $2|S|-1$ and $2|V|-1$. Since the transformation between Steiner tree and our MWCQ problem is polynomial, the MWCQ problem is NP-complete as well.

In the following sections, we study the MWCQ problem by using efficient heuristic algorithms and lower bounds analysis techniques.

4.2 Dynamic Programming based SAID (DP-SAID) Algorithm

In this section, we propose a Dynamic Programming based SAID (DP-SAID) algorithm for the MWCQ problem. With the technique of dynamic programming, DP-SAID can take advantages of the properties of overlapping subproblems to efferently solve the complex MWCQ problem. In specific, we first calculate the solutions of the subproblems with smaller information copies at each node and then combine the solutions of the subproblems to reach an overall solution.

Consider an N-node network having a source node s. Each node u in the networks has a weight w_u, and a quota t_u limiting the number of the information copies this node can retain (or consume). There are a total of R information copies to be delivered from node s to other nodes. We suppose a connected subgraph $S = \{s, u_1, u_2, ...u_r\}(r \leq R)$ is the optimal solution of the MWCQ problem with R information copies, and the nodes are ordered according to their hop distance to the source node. Then the set $S' = \{s, u_1, u_2, ...u_{r-1}\}$ should be the optimal solution to an MWCQ problem with at most $R - t_{u_r}$ information copies and node u_r has direct social connections to nodes in S'. Therefore, we can employ the dynamic programming approach to solve sub-MWCQ problems with smaller R and memorize them for later lookup, thus reducing the number of computations.

In DP-SAID, we first sort the nodes through the Breadth-First-Searching (BFS) algorithm to find nodes' hop distances to the source node. The neighbors of a node are visited according to the weight in non-increasing order. For each node u_i with a order i in BFS, the minimum hop distance to the source node is denoted by d_{u_i} $(1 \leq d_{u_i} \leq R)$. Hence,

to disseminate the content to node u_i, there must exist at lest d_{u_i} content copies in the network. For each content copy number j where $d_{u_i} \leq j \leq R$, the DP-SAID algorithm finds the connected subgraph $S_j(u_i)$ with largest weight, which satisfies: *(i)* $S_j(u_i)$ includes node u_i and the source node s; and *(ii)* the total number of information copies disseminated in $S_j(u_i)$ is at most j. Let $X[u_i, j]$ denote the weight sum of the subgraph $S_j(u_i)$ and $\vec{X}[u_i]$ denote the vector of $X[u_i, j]$ with $d_{u_i} \leq j \leq R$, as in Equation (4.8).

$$\vec{X}[u_i] = (X[u_i, d_{u_i}], X[u_i, d_{u_i} + 1], ..., X[u_i, R]) \tag{4.8}$$

Assume after the BFS algorithm, the sorted node order is $(u_1, u_2, ..., u_i, ..., u_n)$, then $X[u_i, j]$ can be calculated based on $X[u_k, j - t_{u_i}]$ $(k < i)$, by employing the following approach.

(I) if $i = 1$,

$$X[u_i, j] = w_{u_i} (1 \leq j \leq R) \tag{4.9}$$

(II) if $i > 1$,

$$\begin{aligned} &X[u_i, j] = \\ &\max \{X[u_k, j - t_{u_i}] | u_i \text{ directly connects to subgraph } S_{j-t_{u_i}}(u_k)\} \\ &+ w_{u_i} \end{aligned} \tag{4.10}$$

After calculating $X[u_i, j]$ where $d_{u_i} \leq R$ in order, DP-SAID generates an MWCQ solution P as:

$$P = \max\{X[u_i, R] | d_{u_i} \leq R\} \tag{4.11}$$

The DP-SAID algorithm can be described in Algorithm 1.

Figure 4.2 shows an example on how DP-SAID works. In this example, there are 11 nodes including the source node. The weight of each node is the same as its ID and the information quotas (i.e., t_i) are shown in the parentheses. There are $R = 5$ information copies to be delivered from node s to other nodes in the network.

BFS algorithm would visit nodes in the order as $(s, 9, 5, 4, 1, 2, 8, 7, 3, 6, 10)$. As Table

Algorithm 1 DP-SAID Algorithm

Require: Network with N nodes,
Nodes' weights and quotas,
Number of information copies (R).
Ensure: Maximum sum of weight of the nodes who receive the information copies

Sort the nodes as the BFS order $\{u_1, u_2, ..., u_N\}$
for each node u_i **do**
  if $d_{u_i} \leq R$ **then**
    for each j in $d_{u_i} \leq j \leq R$ **do**
      Calculate $X[u_i, j]$ according to Equation (4.9) and (4.10)
    end for
  end if
end for
Find out the maximum $X[u_i, R]$ as the solution

Algorithm 2 TH-SAID Algorithm

Require: network with N nodes,
nodes' weights and quotas,
number of information copies (R).

$S_o = \{s\}$, $S_c = \{n_i | n_i$ is adjacent to $s\}$
The total consumed information quotas T of S_o is initialized as $T = 0$
while $T \leq R$ **do**
  Set the maximum 2-hop WQR $max\_WQR = 0$
  for each node u_i in S_c **do**
    if $T + t_{u_i} \leq R$ **then**
      Calculate the WQR WQR_{u_i} of node u_i
      Find the maximum WQR WQR' of u_i's neighbors who are not in S_o
      $WQR2_{u_i} = WQR_{u_i} + WQR'$
      if $max\_WQR < WQR2_{u_i}$ **then**
        $max\_WQR = WQR2_{u_i}$
        Set the index of the node with maximum $WQR2$ as $Index = i$
      end if
    end if
  end for
  Add node u_{Index} to S_0, and add node u_{Index}'s neighbor nodes to S_c
  $T+ = t_{u_{Index}}$
end while

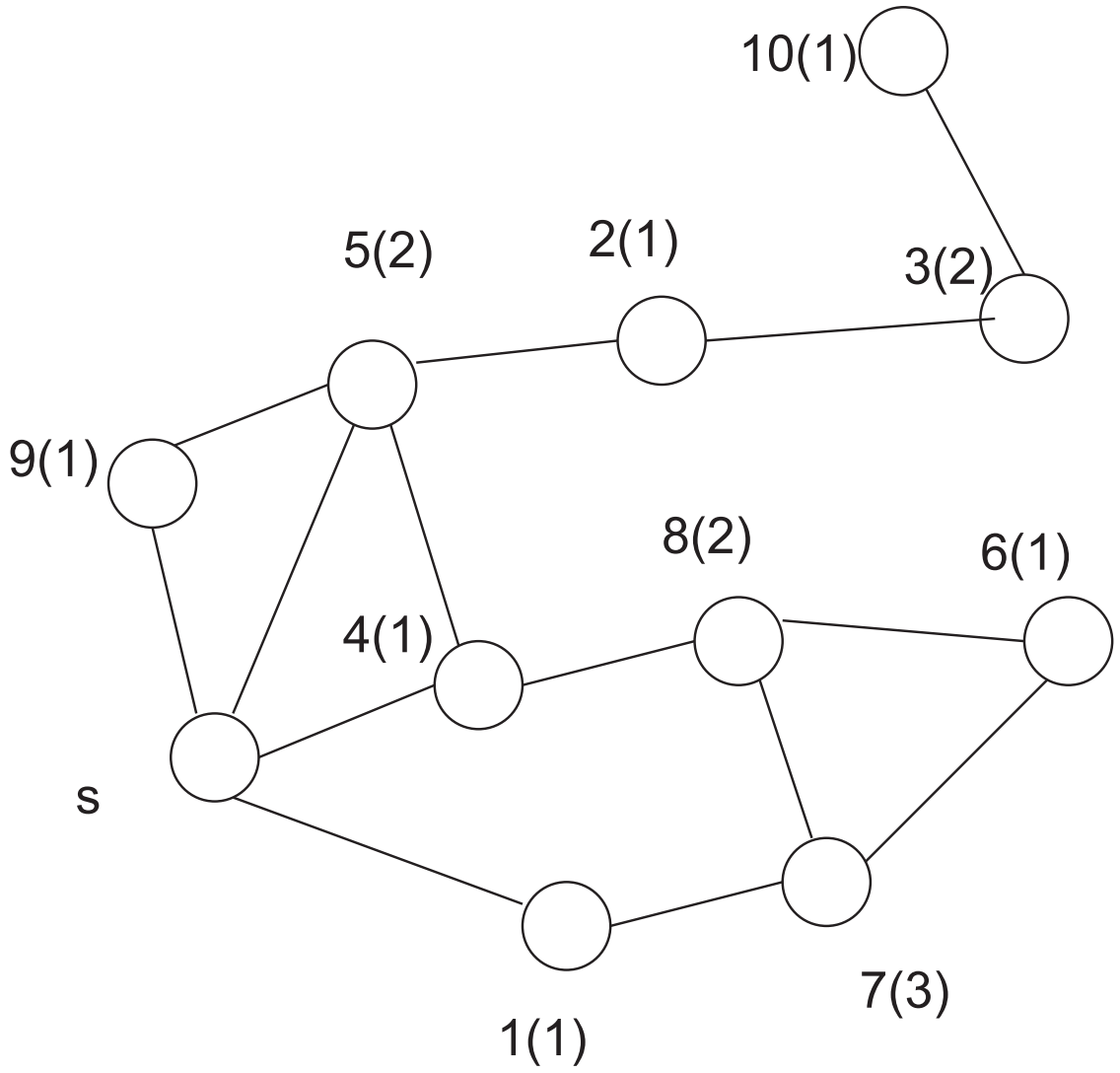


Figure 4.2. An example of DP-SAID algorithm

4.1 shows, the proposed DP-SAID can find the connected nodes set with the largest weight sum. Note that there might be some nodes that is impossible (or unavailable) to be in the solution. For example, node 10 has a shortest path to source node with node $s \to 9 \to 5 \to 2 \to 3 \to 10$. The required copies by node 9, 5, 2 and 3 are 6, which is bigger than the available number of content copies $R = 5$. Thus node 10 can not be included in the solution. The detailed process of the algorithm is listed in Table 4.1, where "x" means unavailable.

Proposition 1. *The DP-SAID algorithm requires* $O(NR^2/2)$ *storage space and* $O(N + E + R^2N^2)$ *running time for a network with* N *nodes,* E *edges, and* R *information copies.*

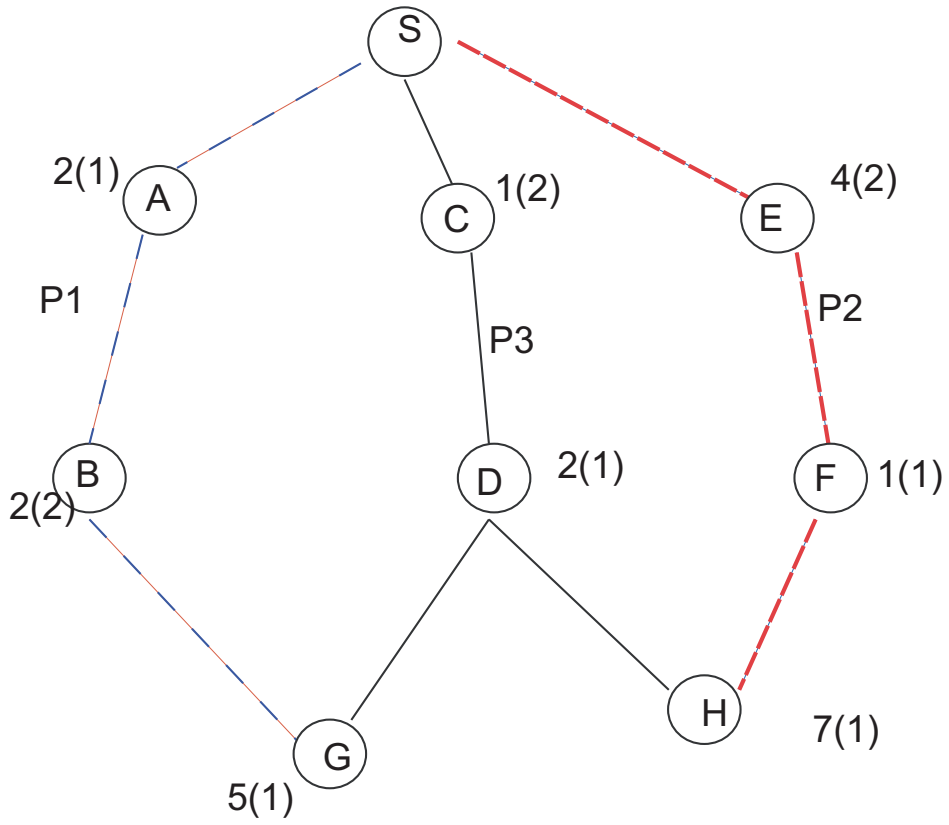


Figure 4.3. An example of shared path structure with $R = 8$

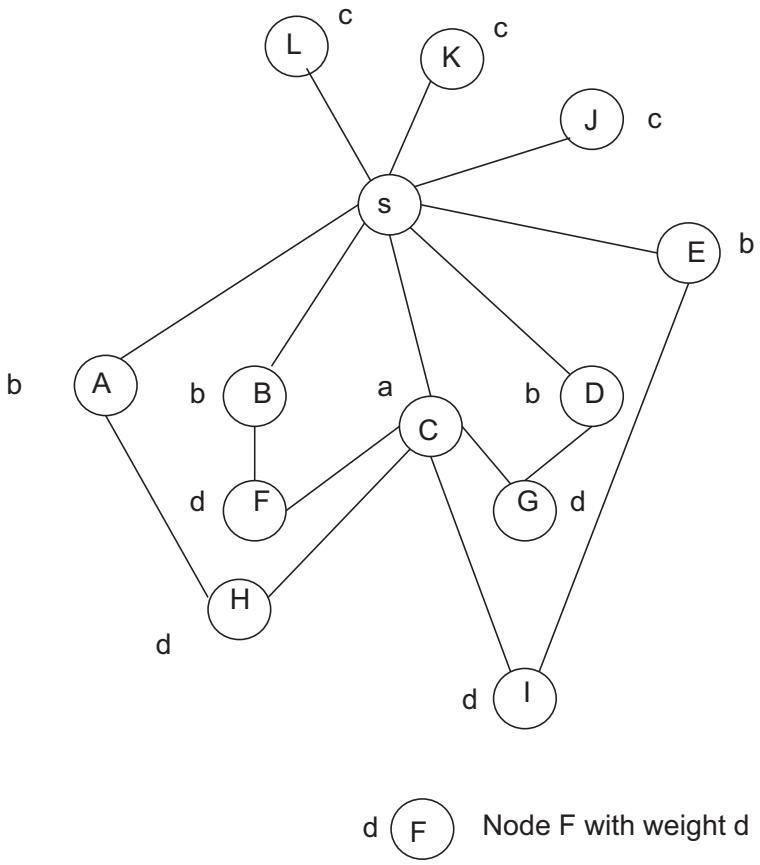


Figure 4.4. A Network structure example of the lower bound case

Table 4.1. The process of DP-SAID algorithm

Node	$X[u,1]$		$X[u,2]$		$X[u,3]$	
	$S_1(u)$	$X[u,1]$	$S_2(u)$	$X[u,2]$	$S_3(u)$	$X[u,3]$
9	$[s,9]$	9	$[s,9]$	9	$[s,9]$	9
5	x	x	$[s,5]$	5	$[s,9,5]$	14
4	$[s,4]$	4	$[s,9,4]$	13	$[s,9,4]$	13
1	$[s,1]$	1	$[s,9,1]$	10	$[s,9,4,1]$	14
2	x	x	x	x	$[s,5,2]$	7
8	x	x	x	x	$[s,4,8]$	12
7	x	x	x	x	x	x
3	x	x	x	x	x	x
6	x	x	x	x	x	x
10	x	x	x	x	x	x
Node	$X[u,4]$		$X[u,5]$			
	$S_4(u)$	$X[u,4]$	$S_5(u)$	$X[u,5]$		
9	$[s,9]$	9	$[s,9]$	9		
5	$[s,9,5]$	14	$[s,9,5]$	14		
4	$[s,9,5,4]$	18	$[s,9,5,4]$	18		
1	$[s,9,5,1]$	15	$[s,9,5,4,1]$	19		
2	$[s,9,5,2]$	16	$[s,9,5,4,2]$	20		
8	$[s,9,4,8]$	21	$[s,9,4,1,8]$	22		
7	$[s,1,7]$	8	$[s,9,1,7]$	17		
3	x	x	$[s,5,2,3]$	10		
6	$[s,4,8,6]$	18	$[s,9,4,8,6]$	27		
10	x	x	x	x		

Proof. Each node maintains storage space for the vector $\vec{X}[u]$, and the corresponding connected subgraph set $S_{j-1}(u_k)$. The total storage space needed for vectors is NR. The total space required to store connected subgraph sets is $O(NR^2/2)$. Thus the space requirement is $O(NR^2/2)$.

The running time complexity is determined by the calculation of the vectors for each node. The computation complexity for the BFS is $O(N+E)$. To calculate $X[u_i,j]$, we only need to search the $S_{j-1}(u_k)$ of previous node u_k that is connected to node u_i. The running time to check the connectivity would be $O(jN)$. So the total running time for node u_i is $O(R^2N)$. Therefore, the time complexity for the whole network is $O(N+E+R^2N^2)$. □

Proposition 2. *The DP-SAID algorithm can achieve a* $\frac{R}{2R-2}$ *lower bound.*

Proof. The DP-SAID algorithm can obtain an optimal solution in most cases through the dynamic programming process. However, DP-SAID maynot yield an optimal solution when multiple dissemination flow paths overlap or share some common intermediate nodes. We call those overlapping structures that cause the DP-SAID solution smaller than the optimal solution, as *shared path structures*. To study the lowerbound of the DP-SAID algorithm, we need to analyze the characteristics of the shared structure and the performance of DP-SAID in the worst case. Figure 4.3 shows an example of such shared structure. In order to disseminate information copies from node s to node G, path $P1 = \{A, B\}$ is the maximum weight path calculated by DP-SAID (denoted by DP-SAID path). Similarly, to deliver information copies to node H, DP-SAID will use path $P2 = \{E, F\}$ as the dissemination flow path. Note that, to disseminate information copies to both node G and H, one can alternatively use the partially shared path $P3 = \{C, D\}$ (denoted by shared path). However, because the weight sum of nodes on $P3$ is smaller than that of $P1$ or $P2$, DP-SAID will greedily employ Path P2 and P3 to deliver information copies to node G and H, respectively. As a result, although $\{A, C, D, G, H, E\}$ is the optimal solution of the MWCQ problem in Figure 3 with 8 information copies, the DP-SAID algorithm would generate the solution as $\{A, B, G, E, F, H\}$ instead. The reason for the sub-optimal results from the DP-SAID in Figure 3 is that DP-SAID cannot take full advantage of the shared path structure in the network. The shared path structure has several characteristics as follows.

First, the weight sum of nodes on shared path $P3$ (defined as $|P3|$) is smaller than the weight sum of nodes on DP-SAID paths $P1$ or $P2$(*i.e.*, $|P3| \leq |P1|$ and $|P3| \leq |P2|$). Second, to include $P3$ in the optimal solution, it must be satisfied that choosing $P3$ and some other nodes yields larger weight sum than $|P1| + |P2|$ (if the same amount of information copies are disseminated). Third, the weights of the shared paths' end nodes (*e.g.*, G and H) should be large enough so that the end nodes need to be included in the optimal solution and DP-SAID solution.

Based on the characteristics of the shared paths, we can derive the worst case to analyze the lower bound of the DP-SAID algorithm. In the worst case, the network consists of the

maximal number of shared paths, the topology of which can be described in Figure 4.4.

As shown in Figure 4.4, in the worst case, there is a shared path ({C}) by multiple end nodes ({F,G,H,I}) to the source node S. The nodes can be divided into four types with four different weight: {a,b,c,d}. a is the weight of the nodes only on shared path(*e.g.*, C), while b is the weight of the nodes only on DP-SAID paths(*e.g.*, A, B). The end nodes on both shared paths and DP-SAID paths(*e.g.*, H, I) have weight d. There are also some additional nodes(*e.g.*, J, K) with weight c who are included in optimal solution but not in DP-SAID solution. The quota of each node is $t_i = 1$ to maximize the difference between the DP-SAID solution and optimal solution. According to the characteristics described above, we have:

$$\begin{cases} b \geq a \\ 2b < a + c \\ c < d \end{cases} \tag{4.12}$$

In the worst case, there are $\frac{R}{2}$ ($R = 2n$, $n \in \mathbb{N}$) nodes with weight d, and $\frac{R}{2}$ nodes with weight b correspondingly. Thus, the optimal solution should be {C,F,G,H,I,J,K,L} with weight sum $(a + \frac{R}{2}d + (\frac{R}{2} - 1)c)$. The solution of the DP-SAID algorithm is {A,H,B,F,D,G,E,I} with weight sum $(\frac{R}{2}b + \frac{R}{2}d)$. Hence, the lower bound of DP-SAID algorithm is:

$$\begin{aligned} LB &= \min \frac{\frac{R}{2}b + \frac{R}{2}d}{a + \frac{R}{2}d + (\frac{R}{2} - 1)c} \\ &\overset{a<b\ll c<d}{>} \frac{\frac{R}{2}d}{\frac{R}{2}d + (\frac{R}{2} - 1)c} + \varepsilon \\ &> \frac{R}{2R - 2} \end{aligned} \tag{4.13}$$

where $\overset{a<b\ll c<d}{>}$ means that in the worst case, nodes with weight d must be included in both the DP-SAID solution and the optimal solution. □

4.3 Two-Hop based greedy SAID (TH-SAID) Algorithm

In this section, we propose a Two-Hop based greedy SAID (TH-SAID) algorithm, which has low running time and space cost comparing to DP-SAID. The algorithm is developed based on the greedy algorithm which adds the adjacent node with largest weight at each step. However, such greedy algorithm may miss some large weight nodes connected to source node through a small weight nodes. On the other side, due to the social influence and social properties in social networks, such small weight nodes that connect two large weight nodes are uncommon. Hence, TH-SAID is developed by investigating the nodes within 2 hops instead of 1 hop. The advantage of this method is to avoid some high weight nodes blocked by a low weight nodes. To take both the information quota and node weight into account, we define a Weight-Quota Ratio (WQR) as the average weight a node can provide per information copy. A higher WQR ratio means a node can contribute more weight while consuming fewer information copies.

As shown in Algorithm.2, we maintain a set S_o as the subgraph set of nodes who have been selected in previous steps. S_o is initialized as $S_o = \{s\}$. A candidate set S_c is also maintained as the neighbor node set of the nodes in S_o. Initially, $S_c = \{u_i | u_i \text{ is a neighbor of } s\}$. At each step, we calculate the 2-hop WQR ratios ($WQR2$) of each node u_i in S_c, which is defined as the sum of node u_i's WQR and the maximum WQR of its neighbor nodes who are not in S_o. Then we add the node not in S_o who has the highest $WQR2$ to S_o at each step. The process recurses until the R content copies are disseminated.

Proposition 3. *The TH-SAID algorithm requires $O(N)$ storage space and $O(R^2D^2)$ running time for a network with N node, average node degree D, and R information copies.*

In this algorithm, we need to store the sets S_o and S_c, which are non-overlapping with each other, so the space cost is $O(N)$. At each step, we need to search $O(RD)$ nodes in the candidate set and $O(RD^2)$ neighbors of those candidate nodes. There are at most R steps, so the running time is $O(R^2D^2)$.

4.4 Performance Evaluation

In this section, we evaluate the performance of the proposed algorithms for the Maximum Weight Connected subgraph with node Quotas (MWCQ) problem. We first examine the performance with different network scales. The running time and impact of R are also evaluated correspondingly.

4.4.1 Network Setting

We set up networks with different network size and connectivity as shown in Table 4.2. The connections between nodes in the networks are randomly generated. The namespace of

Table 4.2. Network Setting

Network No.	Network Size (N)	Average Node Degree (D)
1	100	10
2	500	20
3	1000	50
4	1000	10

information properties and user interests includes 10 topics, each of which has 5 levels. The weights are generated randomly based on the assumption that we have no knowledge about the user profiles. Similarly, we generate the information quotas (i.e. t_i) for each node in the networks with a range $[1, 5]$. My experiments are conducted on a Dell workstation with Intel Xeon E5506 CPU, and 24GB memory. A large number of instances are simulated and the average performance is reported.

My algorithms are compared with straightforward Highest-Weight-First Greedy algorithm (denoted by HWFG). The basic idea of HWFG is that the neighbor node of the result set with the highest weight are selected and added to the result set at each step.

4.4.2 The Sum of Weight Performance

To study the performance of the algorithms upon different amount of information copies and different network scales, we deploy the DP-SAID and TH-SAID algorithms in multiple

networks.

Figure 4.5 plots the sum of weight results in the network with 100 nodes ($N = 100$), average node degree of 10 ($D = 10$). In Figure 4.5, the axis x is the number of information copies (i.e., R) to disseminate in the network, while the axis y represents the weight sum from *DP-SAID*, *TH-SAID* and *HWFG*. As we can see that the DP-SAID algorithm outperforms the TH-SAID and HWFG algorithms, particularly when $R > 20$. This is due to the advantage of dynamic programming in finding possible global optimum. On the other hand, the greedy nature of TH-SAID and HWFG may settle for some local optimized value in the selecting process. TH-SAID algorithm is better than HWFG algorithm because TH-SAID takes into consideration of the two-hops neighbors instead of one-hop neighbors (as HWFG does) in the process of selecting better candidate nodes. In addition, when enlarging the amount of information copies, the performance gain of DP-SAID over TH-SAID and HWFG increases significantly. Similar performance trends can be observed for the networks with $N = 500$ and $N = 1000$ as shown in Figure 4.6 and Figure 4.7, respectively.

4.4.3 Running time

Figure 4.8 shows the running time used for the network with $N = 100$ and $D = 10$. The results for network with $N = 500$ and $N = 1000$ are presented in Figure 4.9 and 4.10, respectively.

Figure 4.8 shows that the overall running time of DP-SAID, TH-SAID and HWFG remains at a low level ($< 60ms$) in small networks. The TH-SAID requires less running time than DP-SAID, but slightly more than HWFG. However, the running time increases rapidly in DP-SAID algorithm when the size of the network increases, as shown in Figure 4.9 and Figure 4.10. Different from DP-SAID, the running time shows a slow increase trend in TH-SAID when the network size increases. Overall, the running time of TH-SAID algorithm is much less than that of DP-SAID, especially when the network is large.

When increasing the information copies (i.e., R), the running time of TH-SAID increases slightly, and keeps close to the HWFG algorithm. The main reason is that the most

costly computation in TH-SAID is the configuration and initiation, which is to detect the connections, gather the weight and quota information. However, the DP-SAID algorithm increases significantly with the number of information copies (i.e., R) as most of the nodes in the network need to calculate the $\vec{X}[u_i]$.

4.4.4 Impact of Network Structure

We also evaluate the impact of network structure on the DP-SAID and TH-SAID algorithms. Networks with different scales present similar results, thus we use the network with 1000 nodes ($N = 1000$) as the evaluation environment. The performance of the DP-SAID and TH-SAID algorithms is shown in Figure 4.11 and Figure 4.12. From the figures, we can see that the weight from both algorithms in networks with average node degrees $D = 50$ slightly bigger than those with average node degree of $D = 10$. Figure 4.13 shows the increase of average node degree can reduce the running time needed for DP-SAID. This is because with more connections, nodes can find out the maximum weighted sets connected to them more quickly. Interestingly, the situation in TH-SAID algorithm is the opposite of DP-SAID, as shown in Figure 4.14. With higher node degree, TH-SAID requires more running time to get the solutions. The reason is that the TH-SAID algorithm needs to detect all the connections in the network, check the weights and quotas of neighbors for each node.

We also evaluate the average node degrees of the connected subgraph solutions. Figure 4.15 shows the results in the network with 1000 nodes and average node degrees $D = 50$. From the figure we can see that TH-SAID algorithm prefers choosing the nodes with higher degrees than DP-SAID algorithm. DP-SAID algorithm emphasizes on what a node connects to instead of how many the node connects to. Besides, with the increase of R, the average node degrees in the found solutions decrease for both algorithms. This indicates that nodes with higher node degrees are more likely to be selected than nodes with lower degrees. In other words, increasing the node degrees in social networks can be helpful in the process of semi-controlled information dissemination.

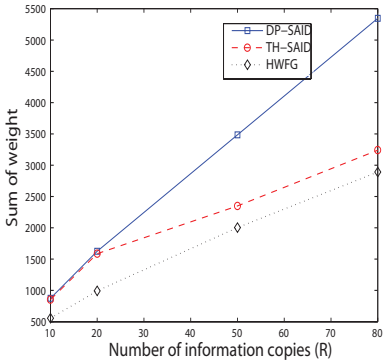


Figure 4.5. Network with $N = 100$, $D = 10$

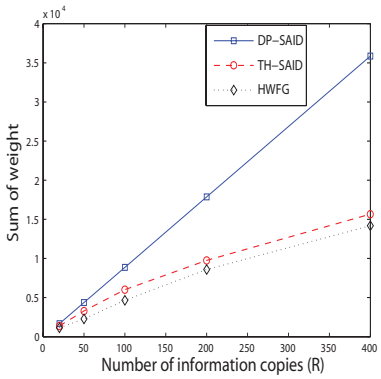


Figure 4.6. Network with $N = 500$, $D = 20$

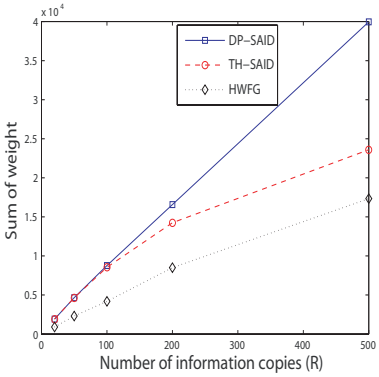


Figure 4.7. Network with $N = 1000$, $D = 50$

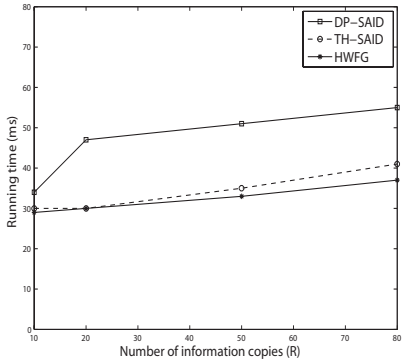


Figure 4.8. Running time with $N = 100$, $D = 10$

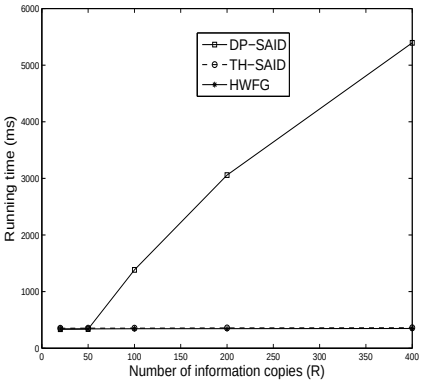


Figure 4.9. Running time of network with $N = 500$, $D = 20$

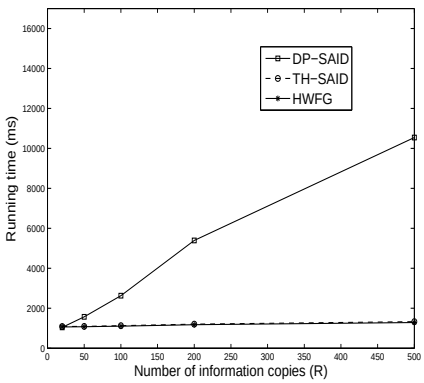


Figure 4.10. Running time of network with $N = 1000$, $D = 50$

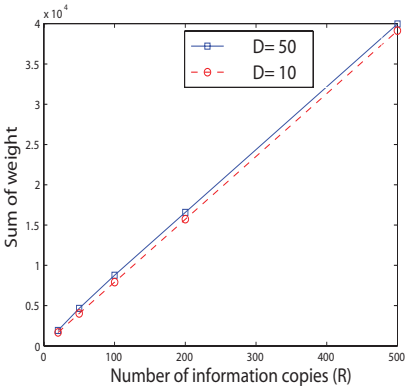


Figure 4.11. DP-SAID on network with $N = 1000$

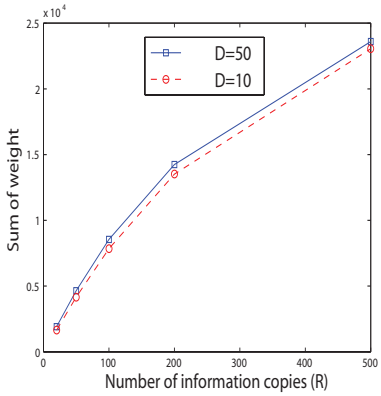


Figure 4.12. TH-SID on network with $N = 1000$

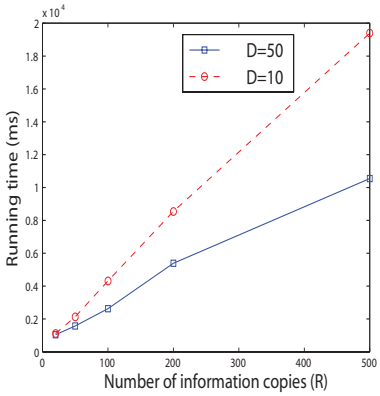


Figure 4.13. Running time of DP-SAID on network with $N = 1000$

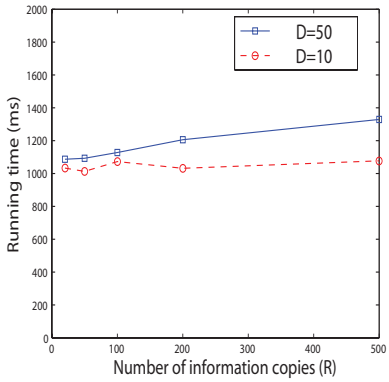


Figure 4.14. Running time of TH-SAID on network with $N = 1000$

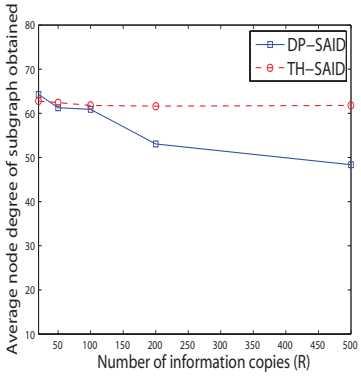


Figure 4.15. Average node degree of the subgraph obtained by algorithms

Chapter 5

CONTENT DISSEMINATION IN OPPORTUNISTIC SOCIAL NETWORKS

The communication among Mobile Social Networks(MSNs) users often happens between opportunistic encounter, which forms Opportunistic Social Networks(OSNs).In the Opportunistic Social Network (OSN), the users can leverage the short range communication technologies such as Wi-Fi [5] and NFC [54] to form an on-the-fly social network. Through the opportunistic contacts among mobile nodes in opportunistic social networks, individual user can share free or self-generated content such as news, pictures and videos. Merchants or organizations are also willing to disseminate their commercial or advertisement content to the interested customers through opportunistic communication of social users.

The intermittent network connectivity and contact uncertainty in OSNs make the content dissemination process unpredictable and difficult. A series of work has studied the data dissemination in OSNs. The users' interests and preference are used for content dissemination in OSNs, as shown in [55][27][32]. In [56]-[26], users with frequent communication or common interests form communities, which are used to find the effective routing for content dissemination. Geography information is used by several studies to help detect the receivers in OSNs [38]. The work in [37] develops the geographic location based geo-community and geo-centrality to model the regularity of users mobility in opportunistic social networks. However, these studies are either only considering the preference/interest of the directly contact users or rely on geography/community information, which may not result in an overall maximized reward for the CP.

In this chapter, we study the authorized content dissemination problem in the Interest-centric Opportunistic Social Network (IOSN). In the IOSN, users move around for the activities or locations they are interested in. For instance, conference attendees move among several sections they are interested in. Hence, if user B has a probability, say p_B, to meet

marketing researchers in the past, the likelihood that user B will meet researchers with similar marketing interest in future is about p_B. In other words, the users with similar interests may have similar trajectories in IOSNs. Based on this observation, we propose the Social Connection Pattern (SCP) to describe the interest distributions of users' social connections. We then develop the Social Connection Pattern based Dissemination (SCPD) algorithm to identify a proper content dissemination strategy when two users contact. The proposed SCPD calculates the maximum expected reward if a certain number of content copies are delivered to a new contactor. Then the SCPD calculates the number of content copies to deliver such that the overall expected reward is maximized. My dataset based simulation shows that the SCPD algorithm is effective and efficient to disseminate the authorized content in interest-centric opportunistic social networks.

There are several unique contributions in our approach. First, our model is built to estimate and predict the overall interest property of the opportunistic connections instead of individual user, which can efficiently avoid the unpredictable opportunistic connections of individual user. Though an individual user may have arbitrary connections, the overall interest property or pattern of the contacted users can be effectively captured in the model. Second, the overhead of our approach is small in terms of storage and computation cost. Each user only need maintain a network size independent small matrix, to record its interest pattern of the connections. The computation cost to calculate and update the SCP is small as well. Third, by predicting the possible interests of future connectors, our social connection pattern based dissemination algorithm novelly and efficiently helps users to calculate the number of content copies to be disseminated so that the total reward is maximized.

5.1 Authorized Content Dissemination in IOSNs

In this section, we introduce the authorized content dissemination problem in IOSNs. In an IOSN, there is a content provider (CP) who generates a limited number of authorized content copies, and disseminates those copies to users during the opportunistic contacts. It is assumed that user retains 1 copy for possible self usage after receiving some content

copies, no matter whether the user is interested in the content or not. And the users are motivated to help disseminate the rest of copies to others in the future opportunistic contacts because of the incentive mechanism provided by the CP. There are many possible incentive mechanism to motivate users in helping disseminate the content, such as virtual check[32], tit-for-tat(TFT)[31]. The work on incentive mechanism is out of scope of this paper. The dissemination process terminates if all users in the networks hold at most one copy. For example, in Figure 5.1, the CP has 4 content copies to disseminate. When D and A meet CP during time t1, the CP deliver 1 and 3 copies to D and A, respectively. Later on (i.e., during time t2), user A rendezvous with B, user A retains one content copy and gives the rest 2 content copies to user B. After retaining one content copy, user B sends the other copy to user C who user B meets at time t3.

There are two types of time-related connection relationship between users: **contactors** and **connector**. User u's contactors are the users who have directly contacted with user u (*e.g.*, A and D are contactors of CP since time t_1 as shown in Figure 5.1). User u's connectors are the users who have not directly contacted with user u but there have been a connection path from those users to u(*e.g.*, B and C are connectors of CP at time t_3). The connectors are further distinguished according to the length of the communication path between users(*i.e.*, the number of **hops**). For example in Figure 5.1, user C is a 3-hop connector of CP since C is connected CP through the communication between CP-A, A-B and B-C. In this work, with no confliction, we also consider the contactors of a user as the 1-hop connectors of the user.

Each user has different interests in the content, which have significant influence on the possible reward obtained by the CP. Generally, the possible that a user adopts or redeems a content copy and generates rewards to the content provider is proportional to the user's interest on the content. hence, the reward amount can be measured as the overall interest of the content receivers. To maximize the overall reward from disseminating the content copies, it is necessary for the CP to take into consideration the interests of both possible contactors and connectors. In Figure 5.1, D is a contactor of CP who is interested in the content, so

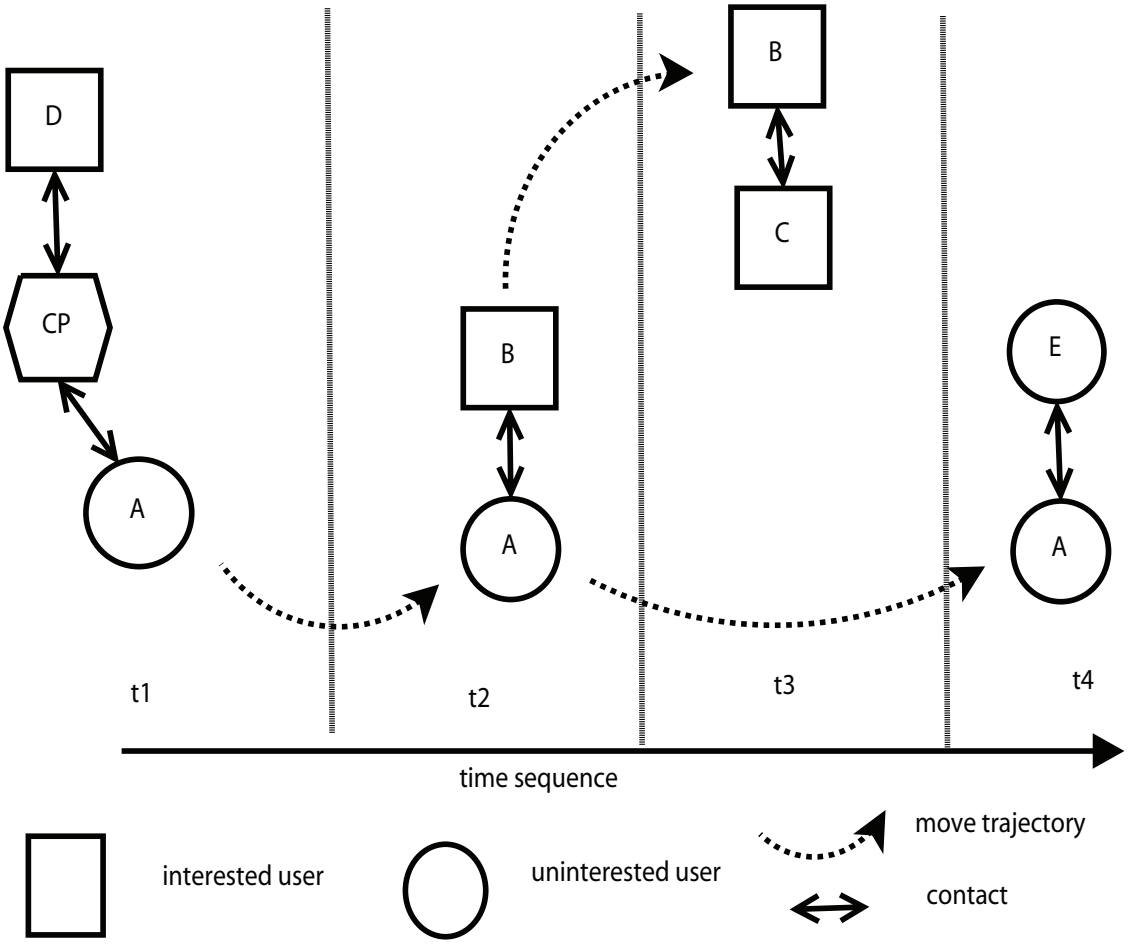


Figure 5.1. An example of authorized content dissemination in IOSNs

CP prefers to deliver a copy to D. On the other hand, user A is not very interested in the content but is able to help disseminate the content to other interested connectors in future (*e.g.*, B and C). Hence one good strategy for the CP is to deliver 1 copy to D, and 3 content copies to A during the contact. User A will retain 1 copy for itself, and deliver the rest of 2 to B and C. User E will not receive any content since it is neither interested in the content nor able to contact with other interested users.

When a content holder who has received a certain number of content from the content provider or other users meets a new contactor, it needs to make its dissemination decision to help maximize the reward generated by the content it holds. Hence, each user who holds a number of content copies has to figure out the following two questions when it meets a new

contactor:

(i) What type of users can the new contactor probably directly contact or indirectly connect in future;

(ii) How many content copies should be delivered to the new contactor so that the overall reward is maximized.

To facilitate the user in answering these two questions and making the dissemination decision, we propose Social Connection Pattern (SCP) and Social Connection Pattern based Dissemination (SCPD) algorithm in the following sections.

5.2 Social Connection Pattern

To calculate the best number of content copies to deliver when a content-holder meets a non-content-holder, we need to find out how possibly the non-content-holder is able to connect the users who are interested in the content. In this section, we introduce the Social Connection Pattern (SCP) to describe the interest distribution of users' contactors and connectors.

The SCP of a users u_i consists of two elements: a social connection pattern matrix P_{it} and a counting vector C_{it}. P_{it} records the interest distribution of the users who have directly contacted or connected through intermediate users with user i till time t. The counting vector C_{it} counts the number of contactors and connectors of user i till time t. The jth row in P_{it} is used to record the interest distribution of the j-hop connectors of user i at time t as shown in Equation (5.1), where N is the largest hop number[1] .

we use **weight** as the measurement of the user's interest in the content, which is corresponding to the reward generated by the user. Accordingly, the value of weight can be obtained by matching the topics of content and users' interested topics. We assume that W is the largest possible weight. In P_{it}, a vector $x_{ijt} = [p_{ij1t}, ..., p_{ijwt}, ..., p_{ijWt}]$ is used to describe the interest distribution of j-hop connectors of user i at time t. p_{ijwt} is the probability

[1]Note that N and the matrix can be simplified when removing the isolated users from the network.

that a j-hop connector of user i at time t has a weight of w.

$$\begin{aligned} P_{it} &= [x_{i1t}, x_{i2t}, ..., x_{iNt}]^T \\ &= \begin{bmatrix} p_{i11t} & p_{i12t} & \cdots & p_{i1Wt} \\ p_{i21t} & p_{i22t} & \cdots & p_{i2Wt} \\ \vdots & \vdots & \vdots & \vdots \\ p_{iN1t} & p_{iN2t} & \cdots & p_{iNWt} \end{bmatrix} \end{aligned} \tag{5.1}$$

According to the "small world" property of opportunistic social networks discussed in [57][58][59], the opportunistic social networks have a high average clustering coefficient and a low average path length. Hence, in most scenarios, N in Equation (5.1) can be set as a small number to reduce the storage cost and computation cost while ensuring the social connection pattern matrix covering most connectable users.

The counting vector C_{it} counts the number of connectors of user i at time t, which describes the potential that user i communicates with others as shown in Equation (5.2):

$$C_{it} = [c_{i1t}, c_{i2t}, ..., c_{ijt}, ..., c_{iNt}] \tag{5.2}$$

where c_{i1t} is obtained by counting the direct contactors of user i while c_{ijt} is the number of j-hop connectors.

When a new user u with weight w_u joins the opportunistic social network (*e.g.*, a new customer entering a shopping mall), the social connection pattern matrix and counting vector is initialized by setting the weight distribution of connectors at each hop uniformly and setting the counting vector to zero. When this user u encounters another user v with weight w_v at time t, they will exchange their social connection pattern matrix and counting vector (P_{ut}, C_{ut}, P_{vt}, C_{vt}). Then the connection pattern matrixes P_{ut} and counting vector C_{ut} are updated by merging the j-hop information in P_{vt} and C_{vt} into the $(j-1)$-hop elements in P_{ut} and C_{ut}, as follows.

(i) For $j = 1$,

$$p'_{ujwt} = \begin{cases} \frac{p_{ujwt} * c_{ujt} + 1}{c_{ujt} + 1} & \text{if } w_v = w \\ \frac{p_{ujwt} * c_{ujt}}{c_{ujt} + 1} & \text{otherwise} \end{cases} \tag{5.3}$$

$$c'_{ujt} = c_{ujt} + 1 \tag{5.4}$$

In Equation (5.3)-(5.4), p'_{u1mt} is the updated interest distribution of user u's contactors and c'_{u1t} is the updated number of contactors of user u at time t.

(ii) For $1 < j \leq N$

$$p'_{ujwt} = \frac{p_{ujwt} * c_{ujt} + p_{v(j-1)wt} * c_{v(j-1)t}}{c_{ujt} + c_{v(j-1)t}} \tag{5.5}$$

$$c'_{ujt} = c_{ujt} + c_{v(j-1)t} \tag{5.6}$$

In Equation (5.5)-(5.6), p'_{ujmt} is the updated interest distribution of user u's j-hop connector and c'_{ujt} is the updated number of j-hop connectors of user u at time t. Similarly, user v will update its P_{vt} and C_{vt}

For each user, the SCP matrix requires storage cost as $O(NW)$, which is independent on the network size and contact frequency. The computation cost for the SCP updating is $O(NW\Theta)$, where Θ is the total number of contacts.

5.3 SCP Based Content Dissemination Algorithm

In this section, we introduce the SCP based content Dissemination algorithm (SCPD) to disseminate the authorized content in IOSNs. Assume there are total M content copies generated by content provider (CP). As mentioned earlier, after the dissemination process terminates, all users in the network have at most 1 content copy. Otherwise, they will continue to disseminate the additional copies to others. The objective is to disseminate those M copies to the users in IOSNs so that the total reward of the users who receive the content copies is maximized.

Suppose user u holds $s+1$ ($M > s \geq 1$) copies of the content and needs to disseminate s

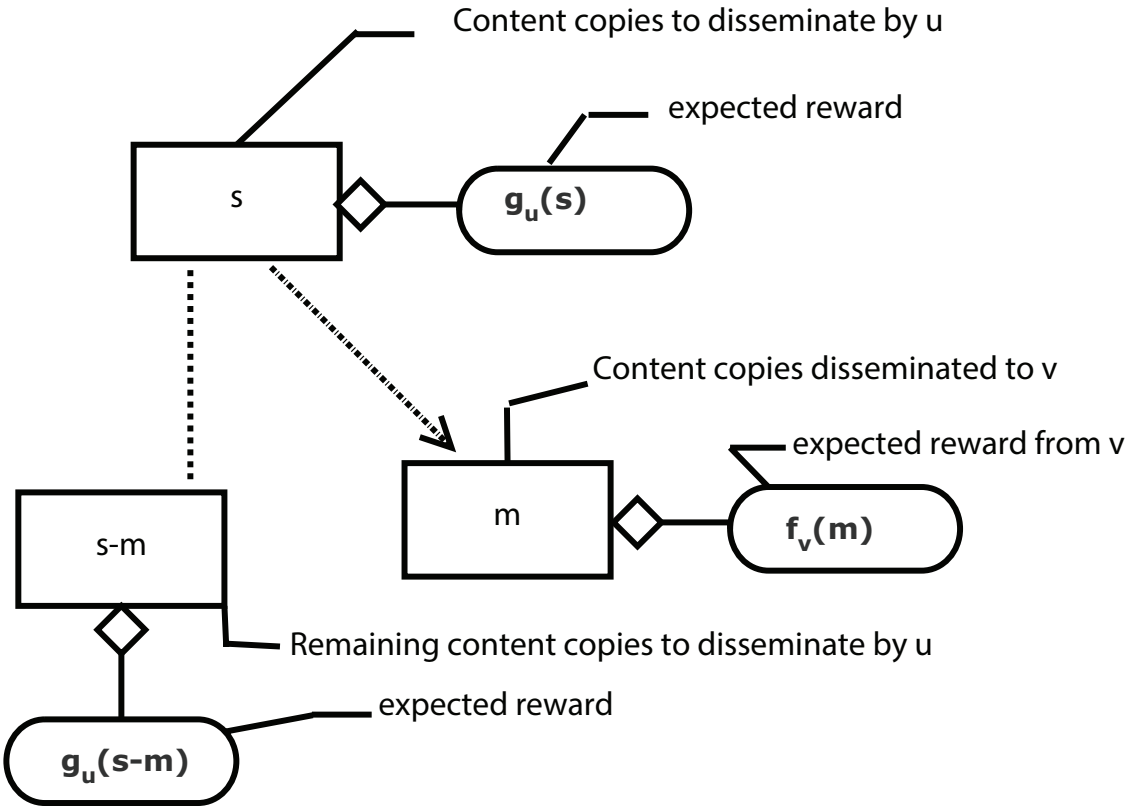


Figure 5.2. Expected reward change during the content dissemination

content copies in the dissemination process starting from u. The s content copies are expected to generate some reward through the dissemination process starting from u, denoted as $g_u(s)$. When user u encounters another user v at time t, user u needs to calculate how many content copies should be delivered to v. Assume user u delivers m content copies to user v, we define $f_v(m)$ as the expected reward to be generated by the m content copies through user v and the dissemination process starting from v. As shown in Figure 5.2, the total expected reward of s content copies is $g_u(s)$ before user u meets user v. After user u disseminates m content copies to user v, the total expected reward of the s content copies now comes from the sum of the $s-m$ copies held by u and m copies held by v, which can be denoted by $g_u(s-m)+f_v(m)$. So user u prefers to disseminate m content copies to user v if the total expected reward of the s content copies increases after the dissemination (*i.e.*, $g_u(s-m)+f_v(m) > g_u(s)$). The basic idea of this SCPD algorithm is to find out the optimal m, denoted by m^* such that the increase of total expected reward is maximized as follows:

$$\begin{aligned} m^* = \quad & argmax_m\{(g_u(s-m)+f_v(m))-g_u(s)\} \\ \text{s.t.} \quad & 0 \le m \le s \end{aligned} \tag{5.7}$$

where m^* is the optimized number of the content copies to be delivered to v by u if u has s content copies to disseminate.

In Equation (5.7), the expected reward of the m content copies disseminated to user v ($f_v(m)$) is calculated based on the SCP of user v, which describes the possible interest of v's potential connectors. The value of $f_v(m)$ can be calculated as:

$$f_v(m) = \max_{m_j} w_v + \sum_{j=1}^{N}\sum_{w=1}^{W} p_{vjwt} \times m_j \tag{5.8}$$

$$\text{s.t.} \sum_{j=1}^{N} m_j \le m-1 \tag{5.9}$$

$$\frac{m_j}{m_{j-1}} \le \frac{c_{vjt}}{c_{v(j-1)t}}, \text{ for all } 1 < j \le N \tag{5.10}$$

$$\forall m_j \in \mathbb{Z} \tag{5.11}$$

where w_v is the interest weight of user v; p_{vjwt} is the element in the social connection pattern matrix P_{vt} and m_j is the number of content copies retained by the j-hop connectors of user v in the dissemination process starting from v. The reward generated by the j-hop connector is $\sum_{w=1}^{W} p_{vjwt} \times m_j$. Hence, Equation (5.8) shows that the total reward created by the m content copies is $w_v + \sum_{j=1}^{N}\sum_{w=1}^{W} p_{vjwt} \times m_j$. Recall that v will retain 1 copy for self usage if it receives any content as shown in Equation (5.9). The constraint in Equation (5.10) is to ensure that there will be sufficient intermediate users to connect to the j-hop connectors. According to the counting vector C_{vt}, the total number of c_{vjt} users in j-hop are connected to user v through $c_{v(j-1)t}$ users in $(j-1)$-hop. Hence, to deliver m_j content to the users in the j-hop, we need the collaboration of at least $\frac{c_{v(j-1)t} m_j}{c_{vjt}}$ users in the $(j-1)$-hop as shown in Equation (5.10).

Similarly, user u has its expectation on the reward from the content copies it holds. If user u holds s content copies, the expected reward $g_u(s)$ is calculated as in Equation (5.12).

$$\begin{aligned} g_u(s) = \quad & \max_{s_j} w_u \sum_{j=1}^{N}\sum_{w=1}^{W} p_{ujwt} \times s_j \\ \text{s.t.} \quad & \sum_{j=1}^{N} s_j \leq s \\ & \frac{s_j}{s_{j-1}} \leq \frac{c_{ujt}}{c_{u(j-1)t}}, \text{ for all } 1 < j \leq N \\ & \forall s_j \in \mathbb{Z} \end{aligned} \tag{5.12}$$

To efficiently resolve this maximization problems in Equation (5.8) and Equation (5.12), we propose the Reward Maximization Algorithm (RMA), which is based on the Branch and Bound algorithm [60]. Take $f_v(m)$ as the example, the sketch of this algorithm is described as follows:

(*I*) **Initialization**: We set $i = 1$ and the value of m_i as 1 to $m-1$ to generate m *possible solutions* s as shown in Line 5-6 in Algorithm. 3. The $f_v(m)$ of each possible solution is

calculated by ignoring the constraint in Equation (5.10). The size of the possible solutions $|s|$ is the value of m_i.

(*II*) **Bound**: As shown in Line 10-12, we check if any possible solution satisfies the constraint in Equation (5.10) as well. If yes, select the possible solution with the largest $f_v(m)$ satisfying the constraint in Equation (5.10) as the *available solution*, and remove any other possible solutions with smaller $f_v(m)$. If the available solution is the only possible solution left, the algorithm terminates; otherwise, go to step (*III*).

(*III*) **Branch**: We set the value of m_{i+1} as 1 to $m - |s| - 1$ to generate new possible solutions, as shown in Line 16-28. The $f_v(m)$ of each new possible solution is calculated by ignoring the constraint in Equation (5.10). The size of the new possible solutions $|s'|$ is $|s| + m_{i+1}$. Update $i = i + 1$ and go to step (*II*) to continue.

With the RWA algorithm calculating $f_v(m)$ and $g_u(s)$, users u can identify the optimal m^* in Equation (5.7) by traversing all m in $0 \leq m \leq s$.

5.4 Simulation and Evaluation

In this section, we present our dataset based simulation and analysis of the proposed schemes.

Two typical opportunistic social network datasets are used in this simulation: infocom2006 [61] and sigcomm2009 [62]. The information about these two datasets is listed in Table 5.1. In this simulation, the reward received by the content provider is represented by the weight sum of the content receivers who retain 1 content copy after the termination of the dissemination process. We use the topics gathered from the participates to calculate the interest weight of a social user on a content. The weights of users are normalized within the range $[0, 100]$. If topic information is missed in the datasets, we randomly generate the topic when needed.

When deploying this Social Connection Pattern based content Dissemination (SCPD) algorithm, we use a number of contact records as the training pool to formulate the social connection pattern for each user. Two different lengths of the training process are adopted:

Table 5.1. Information about the OSN datasets

Name	Nodes	Duration	Contacts	Topics
Infocom	98	4 days	227657	27
Sigcomm	76	4 days	285879	154

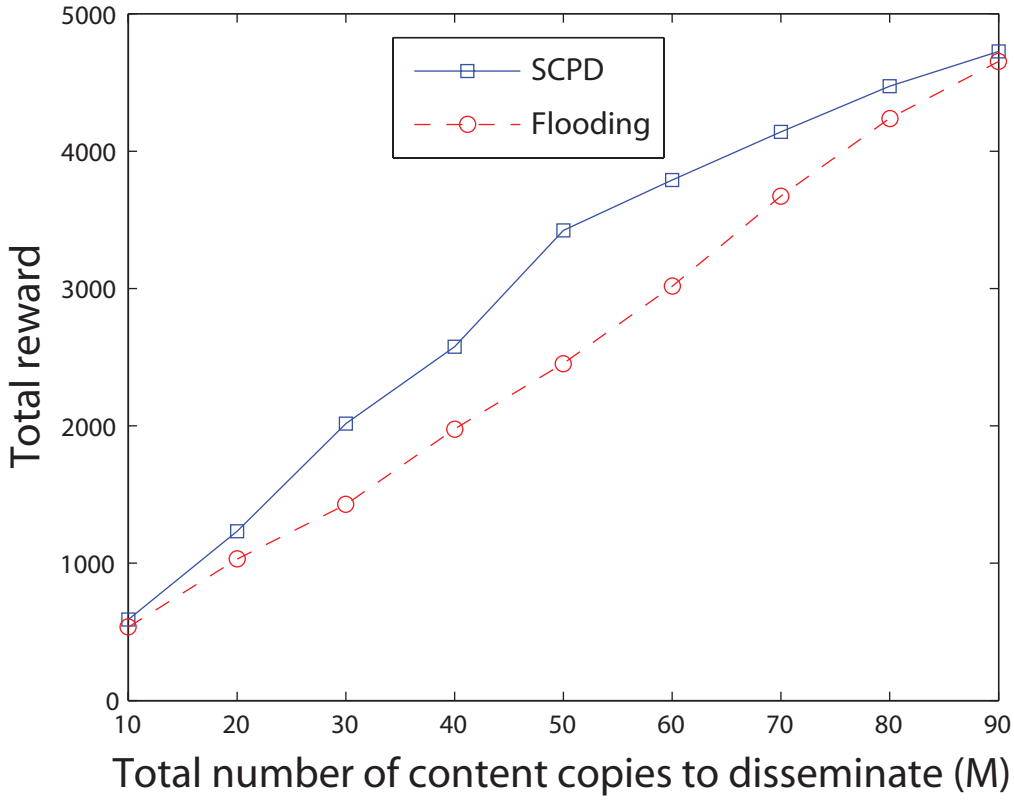


Figure 5.3. Dissemination results of Infocom dataset

10000 contacts records and 30000 contacts records. To evaluate the efficiency of this algorithm, we compare it with the Flooding algorithm [8]. With Flooding algorithm, when a user A meets a new contactor B and B does not obtain the content before, A will deliver half of A's content copies to B, regardless the interest of B.

Figure 5.3 shows the dissemination results of the Infocom dataset. In the simulation, we randomly select a user as the content provider, who intends to disseminate $M(10 \leq M \leq 90$ as shown in the x-axis) content copies to users in the network. The y-axis records the total reward obtained from the content receivers who hold 1 content copy when the dissemination process terminates. The results show that the total reward obtained by the SCPD algorithm

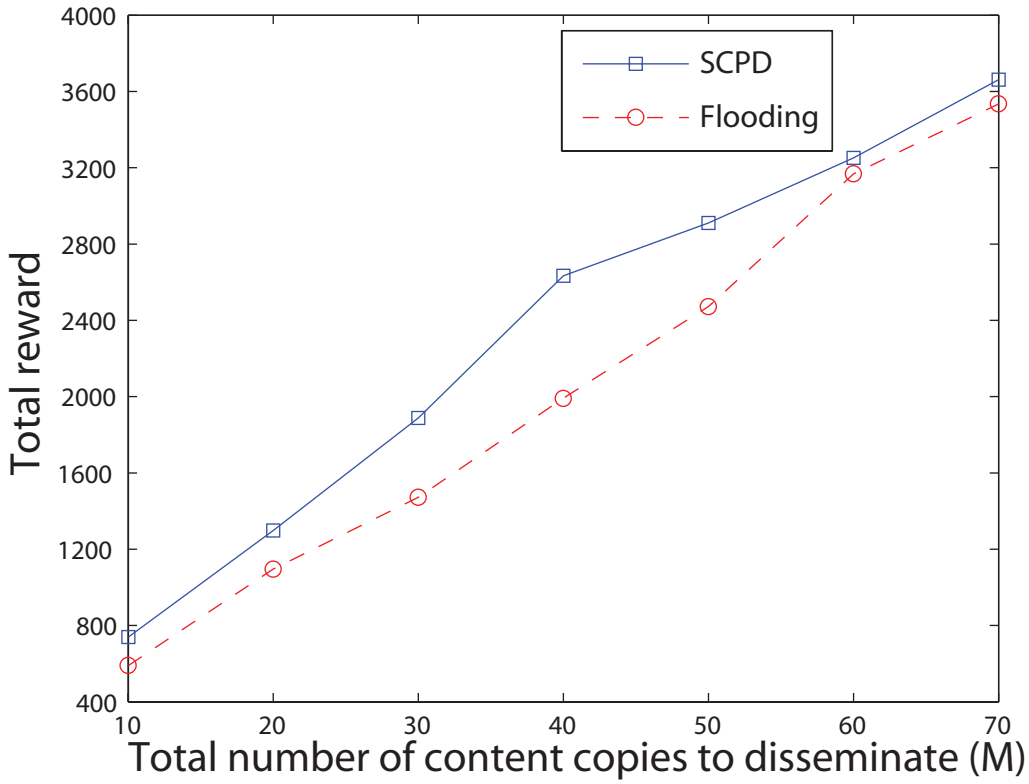


Figure 5.4. Dissemination results of Sigcomm dataset

is significantly higher than the total reward obtained by the Flooding algorithm. In fact, the proposed SCPD can outperform the Flooding scheme as much as 40%. This indicates that SCPD can efficiently disseminate the content to users with higher interest in the content. It also demonstrates that the usage of social connection pattern is able to help predict users' possible connections in future, and help effectively determine the number of content to deliver when two users contact. In Figure 5.3, the SCPD algorithm achieves similar total reward as the flooding scheme when $M = 10$ or $M = 90$ while outperforming the flooding scheme the most when $M = 50$. This can be explained as the follows. When $M = 50$, half of the users in the network will receive the content when the dissemination process terminates. Hence how those users are selected and disseminated content are very important. The SCPD can take advantages of the Social Connection Pattern and user's interest information to maximize the reward achieved. However, when M is small, most users in the network can not receive the

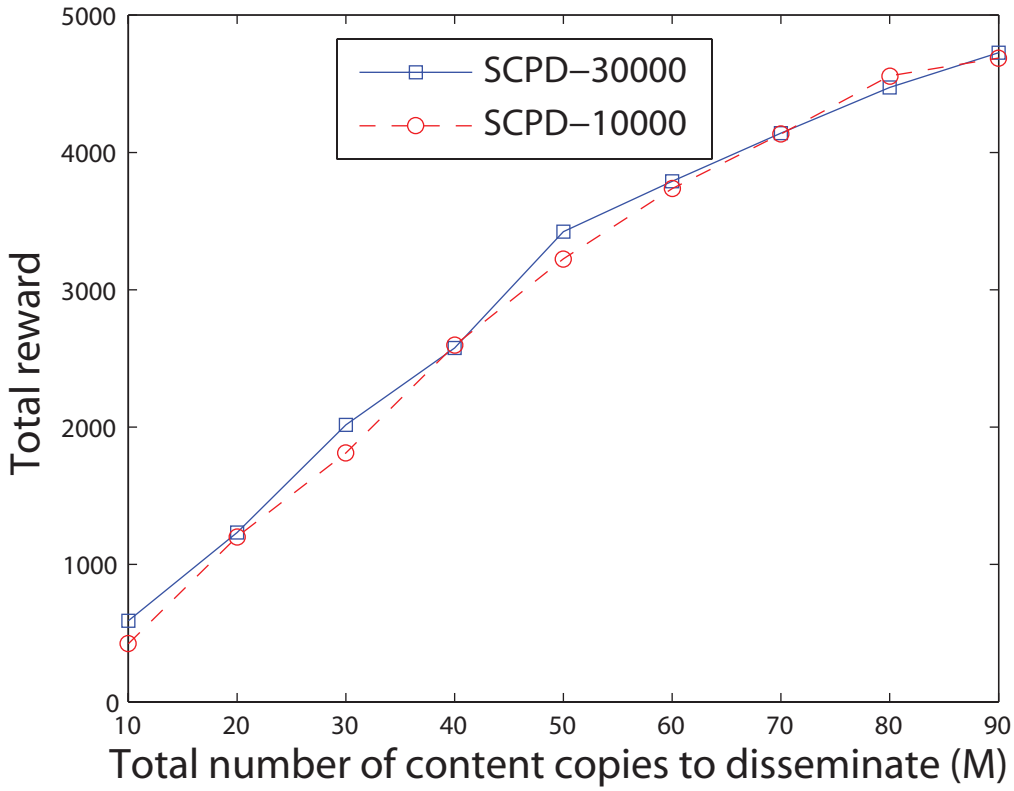


Figure 5.5. Dissemination results of Infocom dataset on training length

content. Even though the proposed SCPD can identify the users with high interest in the content, the interested users may be too far away from the CP (in terms of connection hops). Hence, with a very small number of content copies to disseminate, both algorithms likely deliver them to users who are closer to the CP (,*i.e.*, users who are less number of hops away from the CP). In this case, the social connection pattern and user's interests may not matter much. Similarly, when the number of the content copies is large (e.g., $M = 90$), most of the social users receive the content whereas the social connection pattern and user's interests have little impact on the dissemination process. Thus the performance difference between SCPD and the Flooding algorithm is small when M is very small or very large. Similar conclusions can be found in the results from the Sigcomm dataset, as shown in Figure 5.4.

We also evaluate the performance of the SCPD algorithm with different training length. The training length influences the social connection pattern formulated by each user. By

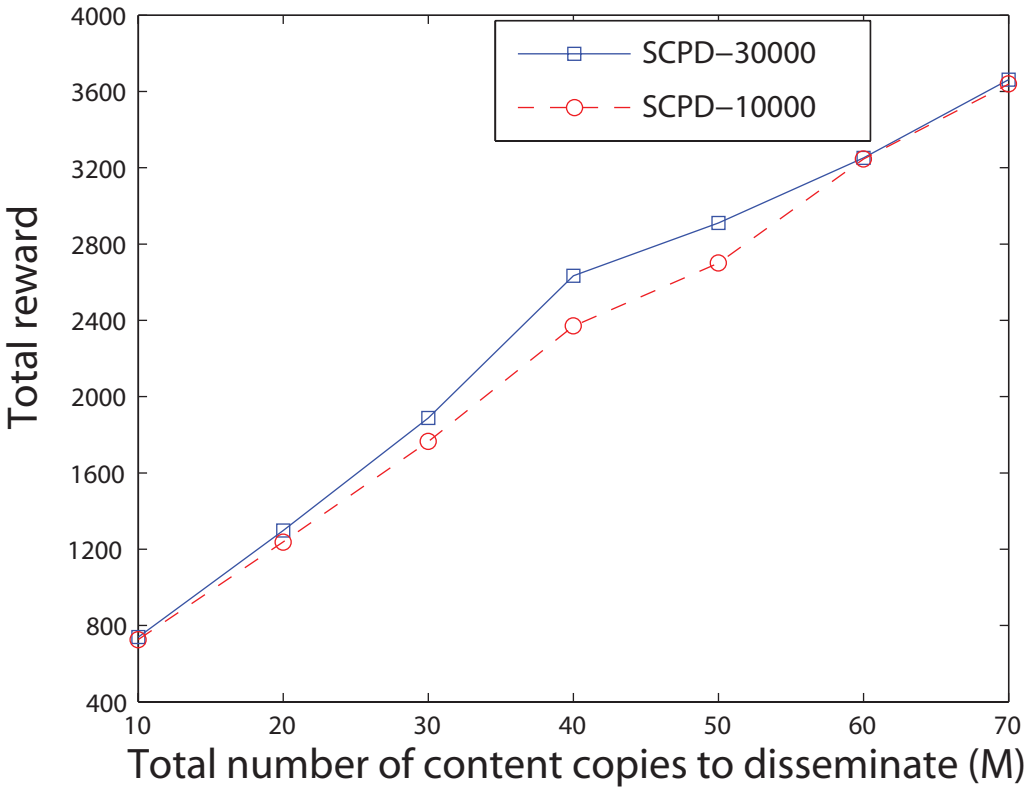
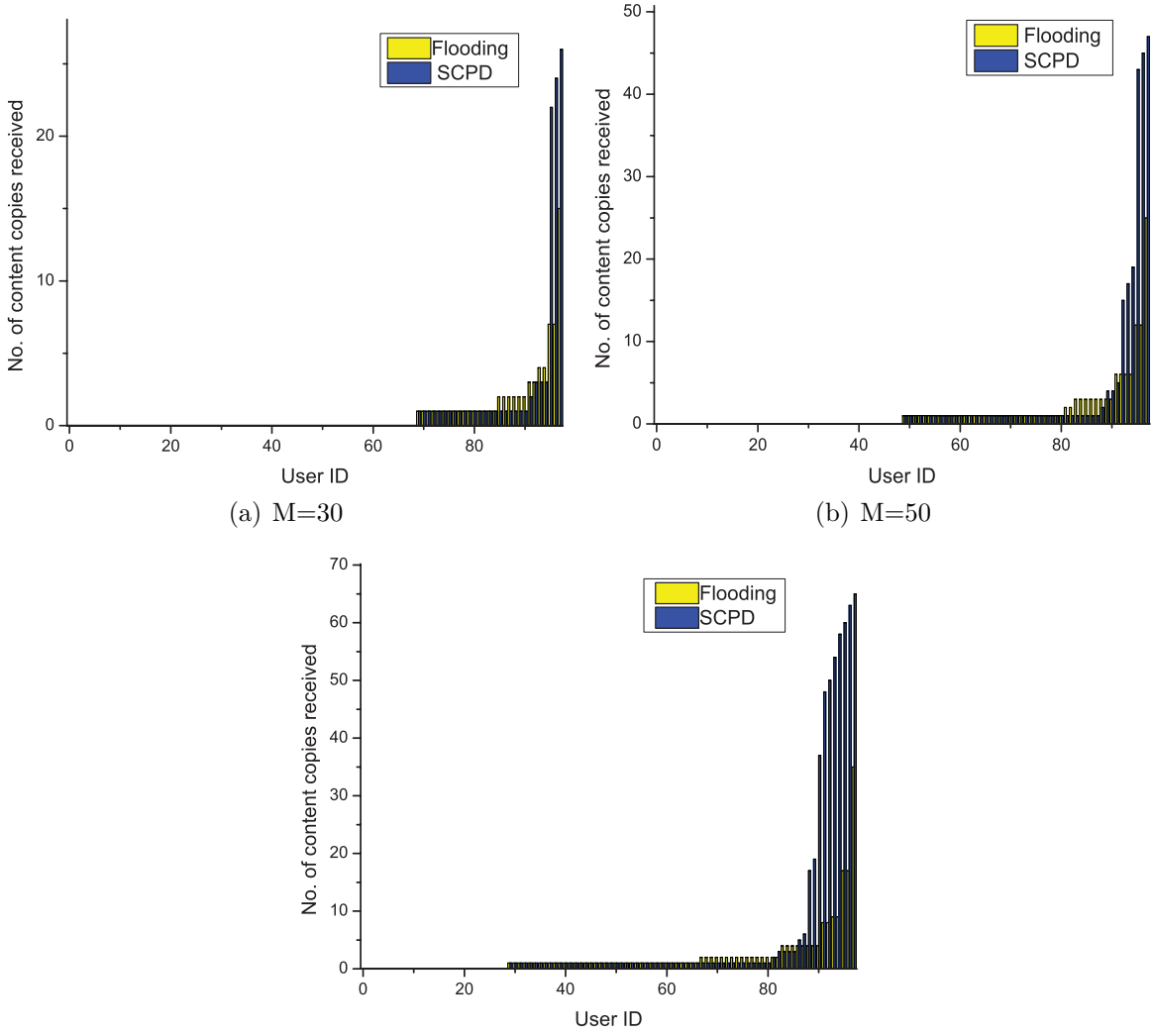


Figure 5.6. Dissemination results of Sigcomm dataset on training length

testing the results with different training length, we can evaluate whether each user actually formulates a stable social connection pattern. The total reward results on different training length of the Infocom and Sigcomm datasets are shown in Figure 5.5 and Figure 5.6 respectively. In the simulation, we set two different length of training: 10000 contact records and 30000 contact records. The results show that the total reward of the content receivers with different training length is quite close to each other for both Infocom dataset and Sigcomm dataset. It means that the social connection pattern of each user after 30000 records training is similar to the pattern formulated in the previous 10000 contact records. Thus the proposed SCPD can reliably predict the future connections according to the previous social connection pattern.

Figure 5.7 shows the distribution of the content copies disseminated to each user. The x-axile is the user ID, which is sorted by the number of content copies received by the user. The

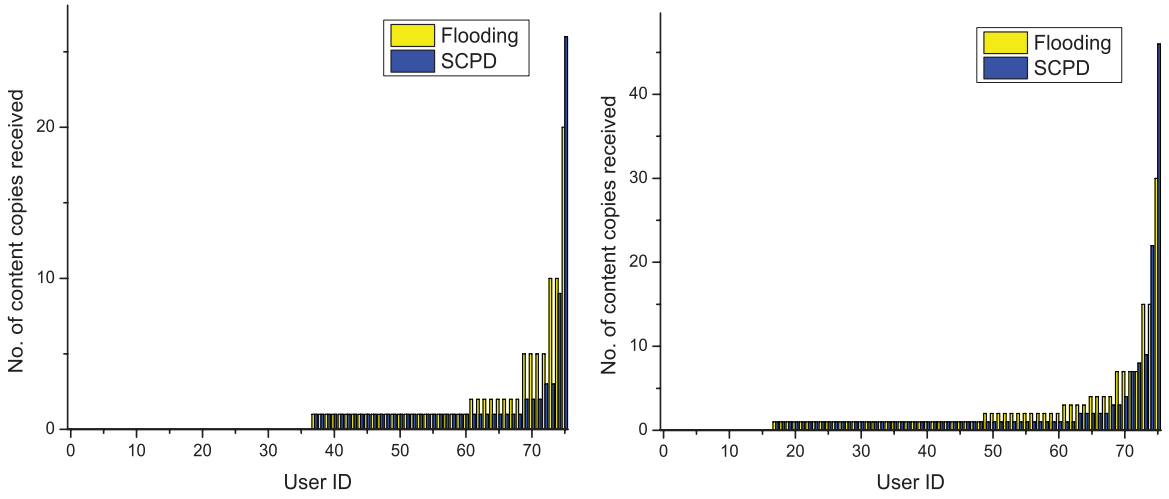


(a) M=30

(b) M=50

(c) M=70

Figure 5.7. Distributions of the received content copies of Infocom Dataset



(a) M=40 (b) M=60

Figure 5.8. Distributions of the received content copies of Sigcomm Dataset

y-axile is the number of content copies received by the user. We compare the dissemination results with different total content copies (*i.e.*, $M = 30, 50, 70$), from the Flooding algorithm and the SCPD algorithm. In the Figure 5.7, the total number of users receiving 1 or more content copies is M as only M users in the network participate the dissemination process. In each figure, there are several critical forwarders who have received a large number of content copies (typically $> \frac{1}{2}M$), and forward those content to other users.

From Figure 5.7 we can find that there are more users receiving a large number of content copies in the SCPD dissemination algorithm than the Flooding dissemination algorithm. The largest number of content copies delivered to users is much larger in SCPD (65 as in Figure 5.7(c)) than that in Flooding algorithm (35 as in Figure 5.7(c)). That indicates that the SCPD algorithm is able to detect the critical forwarders(*e.g.*, the users receiving a large number of content copies) who can efficiently disseminate the content to the interested users.

In the SCPD algorithm, besides the users with a large number of content copies(the critical forwarders) and the users with 0 content copy received, most of the other users have received 1 content copy. It implies that most of users receiving 1 content copy obtains the content copy directly from the critical forwarders with a large number of content copies. However, there are more users in Flooding algorithm receiving more than 1 copies. Those

users act as intermediate forwarders who forward content to others. Hence, the average dissemination path length between the receivers and the source provider in SCPD is smaller than that in Flooding algorithm. The reason is that the SCPD algorithm can efficiently detect the users with high contact probability to the desired users who are interested in the content(*i.e.*, users with high reward). Since the users with high reward have similar interests, they are able to contact some common users with high probability. Hence, the content provider can disseminate a large number of content copies to the common contacted users who can then forward the content to the interested users within a short dissemination path. When there are more content copies to disseminate, as shown in Figure 5.7(b) and Figure 5.7(c), there are more users acting as intermediate forwarders who have received fewer content copies than the critical forwarders. When M is big, we need to disseminate the content to users who may not be contacted by the critical forwarders because of their interests. Hence, we need to rely on some other users as the intermediate forwarders to disseminate the content. As a result, the number of users receiving more than 1 content copies increases.

The evaluation results based on Sigcomm dataset are shown in Figure 5.8. Given the total number of content copies as $M = 40$ and $M = 60$, we calculate the distribution of the content copy number received by users, which are shown in Figure 5.8(a) and Figure 5.8(b) respectively. Figure 5.8 shows similar distribution as that in Figure 5.7, which demonstrates the effectiveness and efficiency of this Social Connection Pattern based Dissemination(SCPD) algorithm.

Algorithm 3 Reward Maximization Algorithm(RMA)

INPUT: The number of content copies m
The Social Connection Pattern matrix of user v P_v
The counting vector of user v C_v
OUTPUT: The maximum reward obtained if user v disseminates the m content copies, denoted by $f_v(m)$
Initialize the possible solution set $L = null$ and *optimal reward solution* $F = 0$
for each $i \in [1, m]$ **do**
  Set a new *possible solution* s as $s = \{m_1, ...m_N\}$, where $m_1 = i$ and $m_j = 0\ \forall j \neq 1$
  Calculate the *maximum reward* of s by ignoring the constraint in Equation (5.10)
  Add s to L
  if {s is an *available solution* satisfying all constraints} AND {the maximum reward of $s >$ *optimal reward solution* F} **then**
    Set F =maximum reward of s
    Remove any possible solution in L with maximum reward smaller than F
  end if
end for
while L contains other *possible solutions* besides the *available solution* **do**
  Select the *possible solution* s' with the largest maximum reward from L
  Find the smallest j with $m_j = 0$ in s'
  Calculate the total number of content copies assigned n in s'
  for $i \in [1, m - n]$ **do**
    Set a new *possible solution* s'' with $s'' = s'$.
    Set the j-hop value m_j in s'' as $m_j = i$.
    Calculate the maximum reward of s''
    if {s is an *available solution* } AND {the maximum reward of $s > F$} **then**
      Set F =maximum reward of s''
    end if
    Remove any possible solution in L with maximum reward smaller than F
  end for
  Remove s'
end while
return F

Chapter 6

USER RECOMMENDATION FOR EFFICIENT CONTENT ACQUIREMENT

In addition to keeping people stay in touch, online social networks have emerged as an important media for information diffusion. The online social network service such as Twitter, Facebook and Instagram, allow users to conveniently acquire and disseminate information such as news, pictures, movie reviews, research publications and reports, through the interaction with their social connections. In these social networks, information is embedded in the posts or microblogs and users acquire information mainly through checking the posts/microblogs (*e.g.*, tweets in Twitter) published by their social connections (*e.g.*, followees in Twitter). Studies have shown that more and more users obtain the information from the social networks[63][64].

When users acquire information from the social networks, two aspects: accuracy and timeliness, are important to the users. The accuracy indicates how attractive is the obtained information, which can be measured by ratio of attractive/interesting information and spam information received by the user. The attractive (or accurate) information may be related to their personal interests, emerging hot topics, popular news or events and so on. When the social connections post/repost information that is not attractive to the user, it brings spam to the user. Meanwhile, a lot of information spreading in online social networks is time sensitive (*e.g.*, news and events). Over a certain period after the information is first published, the information may be worthless to the user. Therefore, for efficient information acquisition, the timeliness is important to measure how promptly a user obtains the needed information. The timeliness can be determined by the interval between the time when the information is generated and the time when a user receives the information. As users obtain information from their connections/followees, the social connections/followees of a user have

a direct impact on what type of information a user obtains and how promptly the information is disseminated to the user (i.e., accuracy and timeliness).

Social networks like Facebook and LinkedIn increase users' social connections by recommending users that share common friends with the target user. The authors in [65] analyze individual's perception of friendship and propose genetic algorithms to recommend high quality and relevant friends. Collaborative filtering (CF) [66][67][68] can recommend objects or other users using the opinion of a set of users.The studies in [69] propose a graph based recommendation system, which analyzes the similarity of users according to their co-tagging behaviors. The work in [70] recommends followees to a user using a randomized method based on the birthday paradox. Similarly, techniques based on Random Walks with Restarts (RWR) [71], Trust [72][73][74], Bayesian inference[75] and location[76][77] have been proposed for friends recommendation.

However, existing work present little contribution on the information acquisition efficiency. These conventional recommendation approaches are mainly based on the social relationships (such as friends, trust, location, colleagues and acquaintances), which may not be optimal for a user to acquire information in a timely and accurate fashion. This is because friends or acquaintances may have different definition on accurate (or attractive) information and they may not obtain/post information in a timely fashion. Similarly, the connections formed according to personal interests[78][65][79] may fail to satisfy users' timeliness requirements. Third, the new connections detected based on local information (such as friend-to-friend, common activity) cannot provide global optimization performance. In addition, these existing work are lack of high volume and complex information processing capacity, which is critically important in current social network services.

In this work, we study how to identify a set of connections such that a user can efficiently acquire timely and attractive information while limiting the information spam[80]. To the best of our knowledge, this is the first work extensively investigating social connection optimization for efficient information acquisition. My contributions can be summarized as follows.

(1) We formally define the problem of optimizing social connections for timely and accurate information acquisition, namely, Social Connection Optimization for efficient Information Acquisition (SCOIA). We prove that the SCOIA problem is NP-complete.

(2) To identify a proper social connection set from a large social system, we propose our distributed information processing approach based on MapReduce model to analyze the information timeliness, accuracy and correlation generated by Big Data environments (Twitter, Facebook, blogs, etc.).

(3) We propose an efficient User Set Selection (USS) algorithm for the SCOIA problem and conduct extensive experiments on a real data set (Twitter Dataset) to evaluate the performance of the proposed algorithm. We demonstrate that our system can significantly improve users' information acquisition accuracy and timeliness while limiting the spam rate.

6.1 Social Connection optimization for Efficient Information Acquisition

In this section, we introduce the proposed framework of Social Connection Optimization for efficient Information Acquisition (SCOIA), which can identify appropriate social connections to a target user to optimize the information acquisition efficiency. Table 6.1 shows the key notations used in the framework.

In a social network N, we use U as the set of user. For a user $u_i \in U$, it may have a need to optimize its social connections so that it can receive and only receive the information it wants from its social connections in timely fashion. So the user u_i would send a request to the social network service provider. The service provider then checks the posting history of u_i and other users in the network to detect and recommend to u_i the best user set that can optimize the information acquisition efficiency of u_i.

For user u_i, P_i represents its history records of the information posts/reposts. P_i can record the posts/reposts within a period of time Γ (e.g., the latest month). Since a user u_i posts or reposts the information mostly because the information is interesting or attractive to u_i, we name the information posted/reposted by u_i as the attractive information for u_i. For each posted/reposted information $p_{ij} \in P_i$, the posting/reposting time is t_{ij}.

Table 6.1. Notations used in SCOIA

U	total user set in a network
u_i	ith user in U
P_i	history record of information posted/reposted by u_i
p_{ij}	jth record of P_i
t_{ij}	posting/reposting time stamp of p_{ij}
L_{ij}	tag list of p_{ij}
x_{mn}	information correlation between posts/reposts p_m and p_n
$\varepsilon(p_m, p_n, \alpha)$	decision on whether p_m and p_n describe the same information
$\varepsilon(p_m, u_i, \alpha)$	decision on whether p_m is interesting to user u_i
α	decision variable for information correlation
$\lvert P_k \bigcap P_i \rvert$	number of posts/reposts of user u_k that are interesting to user u_i
$\lambda_{u_i}(u_k)$	timeliness of u_k on u_i
$\lambda_{u_i}(S)$	timeliness of user set U on u_i
$\delta_{u_i}(u_k)$	support ratio of u_k on u_i
$\delta_{u_i}(S)$	support ratio of user set S on u_i
$\theta_{u_i}(u_k)$	spam ratio of u_k on u_i
$\theta_{u_i}(S)$	spam ratio of user set S on u_i
Λ	timeliness threshold
Θ	spam ratio threshold

The information is identified by the *tags* or *hashtags* in the posts/reposts. For each information $p_{ij} \in P_i$, there is a list of tags $L_{ij} = \{l_{ij1}, l_{ij2}, ...\}$ to describe the content of the information. In the following, we describe how to efficiently identify whether the information of two posts/reposts are correlated.

6.1.1 Information Correlation

The information correlation indicates the similarity of two posts/reposts. By exploring the information correlation between the posts/reposts from two users, we can identify whether the information supplied by a user is interesting or attractive to another user.

Suppose that post/repost p_m has a tag list $L_m = \{l_{m1}, l_{m2}, ...\}$, and post/repost p_n has a tag list $L_n = \{l_{n1}, l_{n2}, ...\}$. The information correlation between p_m and p_n is denoted as x_{mn}, which is calculated as:

$$x_{mn} = \frac{|L_m \bigcap L_n|}{|L_m \bigcup L_n|} \tag{6.1}$$

where $|L_m \bigcap L_n|$ is the number of tags shared by L_m and L_n, and $|L_m \bigcup L_n|$ is the number of tags in the union of L_m and L_n.

Accordingly, we can identify whether two post/reposts are about the same information by adopting a decision variable α as:

$$\varepsilon(p_m, p_n, \alpha) = \begin{cases} 1 & x_{mn} \geq \alpha \\ 0 & x_{mn} < \alpha \end{cases} \tag{6.2}$$

where $\varepsilon(p_m, p_n, \alpha)$ is a heaviside function. Hence, L_m and L_n are about the same information topic when $\varepsilon(p_m, p_n, \alpha) = 1$.

Similarly, we can identify whether a post/repost is attractive/interesting to a user u_i as Equation (6.3).

$$\varepsilon(p_m, u_i, \alpha) = \max_{p_{ij} \in P_i} \varepsilon(p_m, p_{ij}, \alpha) \tag{6.3}$$

Within the framework of SCOIA, the information acquisition efficiency is measured by two features: **timeliness** and **accuracy**. The timeliness measures the time latency for a user to acquire the information. The accuracy is used to differentiate the attractive information from spam provided by user connections. Given a target user u_i, user u_k can provide higher information acquisition efficiency if u_k provides faster information acquisition speed (i.e., higher timeliness), more attractive information and less spam (i.e., higher accuracy). In the following, we introduce our method to calculate the timeliness and accuracy provided by u_k to u_i.

6.1.2 Timeliness Calculation

We define the timeliness of user u_k on user u_i as $\lambda_{u_i}(u_k)$, which measures the time interval between u_k and u_i when they acquire the same information. To calculate the timeliness $\lambda_{u_i}(u_k)$ of user u_k on user u_i, we use the reposting time t_{kj} as the time that user u_k receives the information related to p_{kj}. Next, we need to identify when user u_i receives the same

information. Suppose p_{iq} is the first post/repost obtained by u_i that describes the same information as p_{kj}. The time stamp that user u_i receives the information related to p_{kj} is measured as the posting/reposting time of p_{iq} (*i.e.*, t_{iq}). The acquisition time interval of information p_{kj} between u_k and u_i is calculated as $T_{ki} = t_{kj} - t_{iq}$. A negative value in T_{ki} means u_k receives the information earlier than u_i while a positive value indicates slower information acquisition speed of u_k.

As shown in Equation (6.4), the timeliness $\lambda_{u_i}(u_k)$ provided by user u_k to u_i is calculated as the average acquisition time difference of the information posts/reposts in P_k that are interesting to u_i.

$$\lambda_{u_i}(u_k) = \frac{\sum_{p_{kj} \in (P_k \bar{\bigcap} P_i)} T_{kj}}{|P_i \bar{\bigcap} P_k|} \tag{6.4}$$

In Equation (6.4), $|P_k \bar{\bigcap} P_i|$ is the number of posts/reposts from user u_k, which are interesting to user u_i. It is noted that $|P_k \bar{\bigcap} P_i|$ and $|P_i \bar{\bigcap} P_k|$ have different meanings as they count the information acquired by different users (u_k and u_i, respectively).

We can further calculate the timeliness of a user set S on user u_i as:

$$\lambda_{u_i}(S) = \frac{\sum_{u_k \in S} \lambda_{u_i}(u_k) \cdot |P_i \bar{\bigcap} P_k|}{\sum_{u_k \in S} |P_i \bar{\bigcap} P_k|} \tag{6.5}$$

6.1.3 Accuracy Calculation

Information accuracy is represented by two factors: support ratio (δ) and spam ratio (θ). For a social user $u_k \in U$ with a post/repost record P_k, the support ratio of u_k on u_i ($\delta_{u_i}(u_k)$) is calculated as the percentage of attractive information of user u_i can be offered by u_k, as shown in Equation (6.6).

$$\delta_{u_i}(u_k) = \frac{|P_i \bar{\bigcap} P_k|}{|P_i|} \tag{6.6}$$

The support ratio measures how use u_k can offer the attractive information to user u_i. For example, in Figure 6.1, the posts/reposts from user A cover information $\{a, b, c, d\}$ and User B's posts/reposts cover information $\{a, c, d, e, f\}$. Hence, for user B, 3 attractive

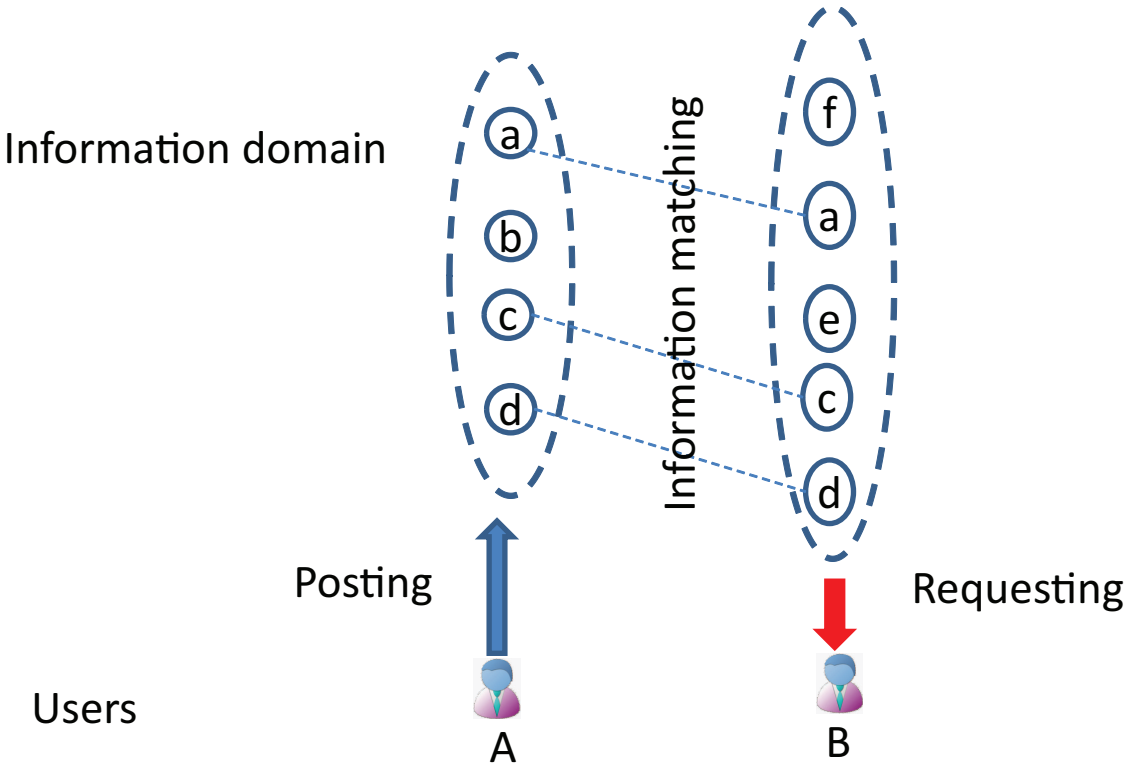


Figure 6.1. An example of support ratio calculation

information (*i.e.*, a, c, d) can be offered by the posts/reposts from user A. Therefore, the potential support ratio of user A to user B is $3/5 = 60\%$.

The spam ratio of u_k on $u_i(\theta_{u_i}(u_k))$ is calculated as the percentage of the information posted/reposted by u_k that is spam to u_i, as shown in the following equation:

$$\theta_{u_i}(u_k) = \frac{|P_k| - |P_k \bar{\bigcap} P_i|}{|P_k|} \tag{6.7}$$

where $|P_k|$ is the total information posted/reposted by user u_k. As the example shown in Figure 6.1, 3 of 4 information offered by user A (*i,e.*, a, c, d) is attractive to user B. Therefore, the spam ratio of user A to user B is $(4-3)/4 = 25\%$.

Similarly, the support ratio and spam ratio of a user set S on user u_i are calculated as:

$$\delta_{u_i}(S) = \frac{\sum_{l:u_l \in S} |P_i \bar{\bigcap} P_l|}{|P_i|} \tag{6.8}$$

$$\theta_{u_i}(S) = \frac{\sum_{l:u_l \in S} |P_l| - \sum_{l:u_l \in S} |P_l \bar{\bigcap} P_i|}{\sum_{l:u_l \in S} |P_l|} \tag{6.9}$$

where P_S is the total information posted/reposted by the users in S.

Note that more accurate information does not derive higher support ratio. If two or more posts in user A cover the same information, user B will receive the same amount of attractive information from user A. Hence, user A contributes the same support ratio to user B. The attractive posts which do not contribute on the support ratio is *redundant information* in that case.

6.1.4 MapReduce-based Information Processing

The huge size of users and records data make the information processing extremely challenging. Conventional sequential information processing approaches have high demands for the storage and computation capacity of servers, making it difficult to be deployed in practice. In this paper, we propose a nested MapReduce-based information processing approach to provide efficient and fast computation.

MapReduce is an efficient framework for parallel processing huge dataset by using a large number of computing nodes. In a Mapreduce program, there are major two steps for the processing: "Map" step and "reduce" step. In Map step, a data block is send to a Map node who applies the Map() function. The Map() function matches each datum, and outputs corresponding key-value pairs. The output of Map() function is shuffled so that the pairs with same "key" are located on the same Reduce nodes. The Reduce nodes then load the Reduce() function on the <key, value> pairs with same "key".

In the nested MapReduce scheme, we do two Map-Reduce steps. At the beginning, the user post/repost records will be sent to the Map nodes, who will call the Map() function. In the Map() function, we take the information attractive to the target user as the key. Then the Map() function will generate a <info_ID, (user_ID, time)> pair for each record. In the pair of <info_ID, (user_ID, time)>, *info_ID* is the index of the information attractive to the target user; *user_ID* is the ID of the post record and *time* is the posting time of the post.

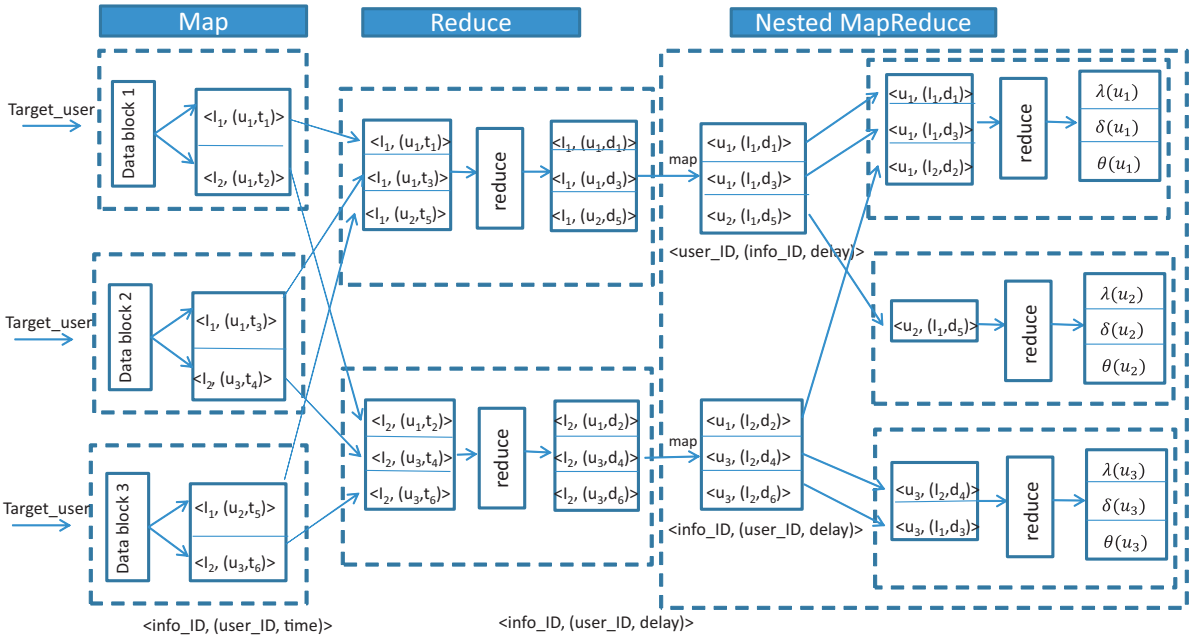


Figure 6.2. The nested MapReduce algorithm

After the Map process, all records covering the same information (i.e., with the same "key") will be stored in the same storage block and sent to the reduce nodes. A reduce node then will do Reduce() function on a single storage block. In Reduce() function, we compare the posting time of the post/repost with same "key" value to calculate the posting delay of each user on the same information. As a result, the Reduce() function will output a triplet < info_ID, (user_ID, delay) > for each record <info_ID, (user_ID, time)>. In the triplet, the delay is the calculated as the difference between the posting time of *user_ID* and the posting time of the target user's record on *info_ID*. For the delay of the records with the key "spam", the Reduce() function returns the triplets of every post/repost and the *delay* in the triplets is set as 0.

The Reduce() function will further call a nested MapReduce program to calculate the accuracy and timeliness. In this step, the data to be processed is the triplet output from previous Reduce() function, and the user_ID is selected as the key. In the nested Map() function, the structure of the <key, value> pair is defined as <user_ID, info_ID,delay>. We map the triplet with same user ID to the same data block and deliver the data block to

Reduce() function. The Reduce() function will calculate the timeliness $\lambda_{u_o}(u)$, support ratio $\delta_{u_o}(u)$ and spam ratio $\sigma_{u_o}(u)$ for each *user_ID* based on Equation (6.4), (6.6) and (6.7), respectively.

6.1.5 Social Connection Optimization for Efficient Information Acquisition(SCOIA)

The problem of Social Connection Optimization for efficient Information Acquisition (SCOIA) can be defined as: given a social network user set U with the post/repost history of all users, how to find the most appropriate user set S that provides the highest support ratio $\delta_{u_o}(S)$ for a target user u_o while satisfying the following two constraints: (i) the spam ratio from S is less than the spam threshold, and (ii) the information receiving latency is under the timeliness threshold. The SCOIA problem can be formulated as:

$$\begin{aligned} & \max_{S \subset U} \delta_{u_o}(S) \\ \text{subject to:} \quad & \lambda_{u_o}(S) \leq \Lambda \\ & \theta_{u_o}(S) \leq \Theta \end{aligned} \tag{6.10}$$

where Λ is the maximum information receiving latency and Θ is the maximum spam rate allowed.

The value of Λ and Θ can be *fixed threshold* or *adaptive threshold*. For the fixed threshold, the users themselves set fixed values for Λ and Θ, representing the personal tolerance on the information acquisition delay and spam rate. The adaptive thresholds allow the values to be changed according to the current information acquisition latency and spam ratio of user u_o.

Theorem 2. *The SCOIA problem is NP-complete*

The SCOIA problem can be converted to Knapsack problem [81], which is proved as NP-complete problem. Given a set of items, each with a mass and value. The Knapsack problem is defined to determine a number of items so that the total weight is less than or equal to a given limit and the total value is as large as possible. For the SCOIA problem, we set the

timeliness threshold Θ as infinite. In that way, the SCOIA problem can be converted to a Knapsack problem, in which Λ represents the weight limit and the total value is represented by $\delta_{u_o}(S)$. As a special case of SCOIA problem, the Knapsack is NP-complete. Hence, the SCOIA problem is NP-complete.

6.2 User Set Selection (USS) Algorithm

Algorithm 4 User Set Selection (USS) Algorithm

Require: network with U users,
target user u_o,
nodes' post/repost record,
spam ratio threshold Θ,
timeliness threshold Λ.
Ensure: user set S_o with maximal support ratio on u_o
calculate $\lambda_{u_o}(u)$, $\delta_{u_o}(u)$ and $\theta_{u_o}(u)$ for each u by using the nested Mapreduce algorithm
max_support_ratio = 0;
find out user u_i satisfying the spam ratio constraint and timeliness ratio constraints
add u_i to user set S
add S to candidate solution set C
max_support_ratio = $\delta_{u_o}(S)$
while C is not empty **do**
  for all c in C **do**
    for all u in $U - c$ **do**
      if $\{c \bigcup u$ satisfies $\theta_{u_o}(c \bigcup u) \leq \Theta$ and $\lambda(c \bigcup u) \leq \Lambda\}$ and $\{c \bigcup u$ not in $C\}$ **then**
        add $c \bigcup u$ to C
        if $\delta(c \bigcup u) >$ max_support_ratio **then**
          max_support_ratio = $\delta(c \bigcup u)$
          $S_o = c \bigcup u$
        end if
      end if
    end for
    remove c from C
  end for
end while

Considering the non-linear constraints in the SCOIA problem, the conventional schemes for Knapsack problem such as Dynamic programming algorithms, can not be directly applied. In this section, we propose User Set Selection (USS) algorithm to efficiently solve SCOIA.

The algorithm is developed based on the following theorem.

Theorem 3. *If all users in a user set S ($\forall u_i \in S$) satisfy $\theta_{u_o}(u_i) \leq \Theta$ and $\lambda(i) \leq \Lambda$, then $\theta_{u_o}(S) \leq \Theta$.*

Similarly, we have Theorem 3, whereas, if any user in a user set S ($\forall u_i \in S$) satisfies $\lambda(i) \leq \Lambda$, then $\lambda(S) \leq \Lambda$.

Theorem 4. *If all users in a user set S ($\forall u_i \in S$) satisfy $\theta_{u_o}(u_i) \leq \Theta$ and $\lambda(i) \leq \Lambda$, then $\lambda(S) \leq \Lambda$.*

Based on the above two theorems, we propose the User Set Selection (USS) algorithm as shown in Algorithm 4. As shown in Line 3-6 of Algorithm 4, USS first selects users who satisfy the constraints $\theta_{u_o}(u_i) \leq \Theta$ and $\lambda(i) \leq \Lambda$, to be added into user set S, which guarantees that $\theta_{u_o}(S) \leq \Theta$ and $\lambda(S) \leq \Lambda$ according to the theorems. Then we have S_k as the user set in step k.

Then, for each user u_i in $U - S$, USS generates a new user set $S_1 = S \bigcup \{u_i\}$ at the size of $|S| + 1$. If S_1 satisfies the constraints $\theta_{u_o}(S_1) \leq \Theta$ and $\lambda(S_1) \leq \Lambda$, S_1 is called a candidate solution with a size of $|S| + 1$. USS will add S_1 to a candidate set C_1, in which all candidates have a size of $|S| + 1$.

Recursively, from the candidate S_k in C_k, USS generates a new user set S_{k+1} with a size of $|S| + k + 1$ by combining S_k with a user who is not in S_k. If S_{k+1} satisfies the constraints $\theta_{u_o}(S_{k+1}) \leq \Theta$ and $\lambda(S_{k+1}) \leq \Lambda$, S_{k+1} is a new candidate solution. USS will add S_{k+1} to a candidate set C_{k+1}, in which all candidates have a size of $|S| + k + 1$. Repeat this process as shown in Line 7-20 until no more candidate is generated. At the end, the solution of the user selection problem is the candidate set with the highest support ratio.

6.3 Performance Evaluation

In this section, we present our dataset based evaluation. We first trace Twitter's posting/reposting history by using Tweepy [82]. We select 1000 active Twitter users randomly

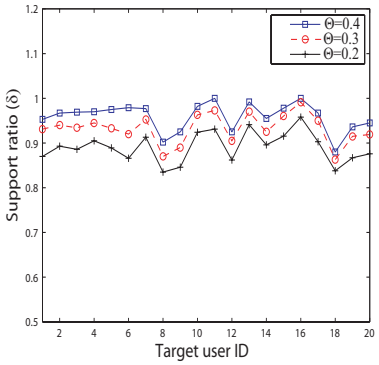


Figure 6.3. Support ratio of selected users

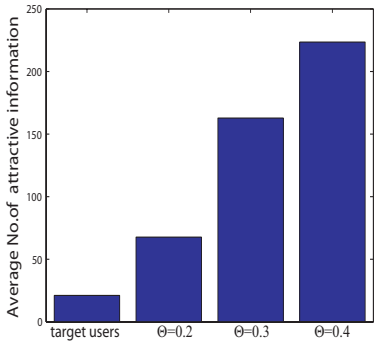


Figure 6.4. Number of attractive information offered

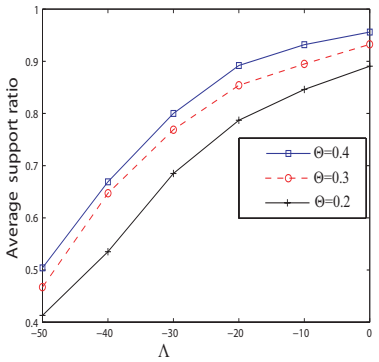


Figure 6.5. Average support ratio of selected connections

and extract their posting/reposting history between 03-01-2015 and 04-01-2015. We obtain 101509 records in the dataset. In the experiment, 20 users are randomly selected as the target users. The records within the first 15 days (*i.e.*, the records between 03-01-2015 and 03-15-2015) are selected as the training set to calculate the information correlation and information acquisition efficiency parameters (i.e., λ, θ and δ). Based on those parameters, the User Set Selection (USS) algorithm is employed to identify user set S, which is selected to optimize social connections for the target users. The rest records between 03-16-2015 and 04-01-2015 are used as the testing set to evaluate the connection optimization approach.

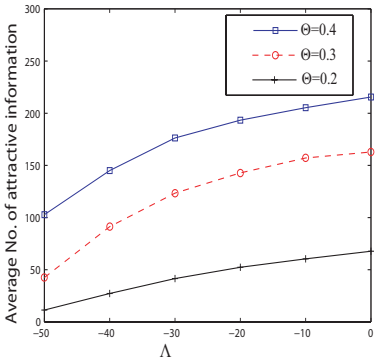


Figure 6.6. Average number of attractive information offered

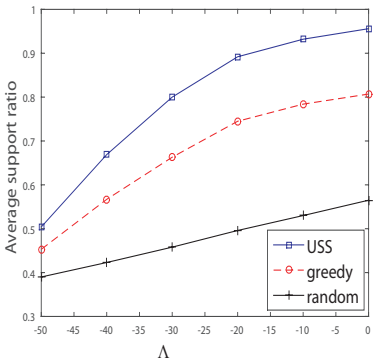


Figure 6.7. Average support ratio of selected connections

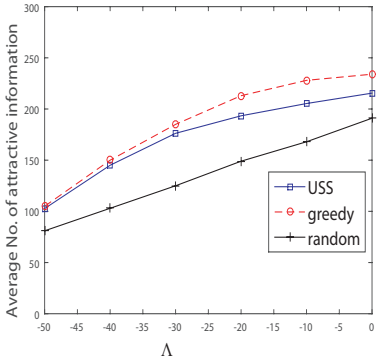


Figure 6.8. Average number of attractive information offered by selected connections

6.3.1 Spam Ratio Threshold

To evaluate the performance of spam ratio threshold Θ, we adaptively choose the timeliness threshold Λ. Three different Θ are tested: $\Theta = \{0.2, 0.3, 0.4\}$. For each Θ, we calculate the selected user set S for a given target user u_o, and record the support ratio of S on u_o (i.e., $\delta_{u_o}(S)$) as well as the number of attractive information offered by S to u_o.

Figure 6.3 shows the support ratio performance of the proposed USS. The x-axis is the ID of the target users and y-axis is the support ratio of the selected users on the target users. It is shown that most of the selected users present high support ratio (above 85%) to the target users. This indicates that our social connection optimization framework is efficient to identify users who can offer attractive information. When the spam ratio threshold increases

from $\Theta = 0.2$ to $\Theta = 0.4$, the support ratios of the selected users increase as well. This is because USS can detect more users with higher spam ratio threshold, resulting more attractive information.

We then record the average number of attractive information from the selected users by USS, as shown in Figure 6.4. In Figure 6.4, the y-axis is the average number of attractive information from the selected users and the x-axis is the spam ratio threshold. The first bar along the x-axis is the average number of information post/repost by the target users in the training phase, which provides a baseline for the comparison. From Figure 6.4, we can see that the selected users are able to provide more attractive information to the target users when Θ is larger. This is because bigger Θ allows the target users to tolerate more spam. Then USS will be able to identify more connections for the target users, leading to more attractive information obtained by target users.

6.3.2 Timeliness Threshold

To evaluate the impacts of the timeliness threshold, we set the timeliness threshold Λ in the range of $[-50, 0]$. The threshold $\Lambda = -50$ requires that the selected users/connections acquire information at least $50h$ earlier than the target user on average while $\Lambda = 0$ means that the selected users/connections have similar information acquisition speed with the target user. Three spam ratio thresholds are tested (i.e., $\Theta = \{0.2, 0.3, 0.4\}$).

In Figure 6.5, the y-axis shows the average support ratio of the selected users from USS and the x-axis is the timeliness threshold. We can see that a larger timeliness threshold yields a higher support ratio from the proposed USS. The reason is that more users are selected when a larger timeliness threshold is chosen. In other words, if the target users have higher tolerance on the information acquisition delay, more likely the target user will receive more attractive information. This is further verified in Figure 6.6, where the y-axis denotes the average number of attractive information provided by the selected users and x-axis represents the timeliness threshold.

6.3.3 Algorithms Comparison

We compare our social connection optimization algorithm USS with other user selection strategies. Two typical strategies are deployed as the baselines for the comparison: greedy selection algorithm and random selection algorithm, denoted by " greedy" and "random" in Figure 6.7 - 6.8. The greedy selection algorithm repeatedly selects the user providing the most attractive information while satisfying the timeliness and spam ratio constraints from the remaining connection pool, and ends till no more users can be selected. The greedy algorithm represents the metrics of interest-based social connection recommendation schemes as in [78][65][79]. The random selection algorithm randomly selects a user satisfying the timeliness and spam ratio constraints at each iteration until no more users can be selected.

Figure 6.7 shows the average support ratio of the selected connections from different selection strategies. In the simulation, the spam ratio threshold (Θ) is set as 0.4. The results show that our USS algorithm can achieve better support ratio than the greedy and random selection algorithms because the USS algorithm can identify the globally optimized social connection set. When comparing the average attractive information provided by the selected connections in Figure 6.8, however, it is interesting to see that the greedy selection algorithm can provide more attractive information than the USS algorithm. In other words, when compared to USS, the greedy selection algorithm can provide a larger number of attractive information while yielding a lower support ratio. This is because the users selected by the greedy selection algorithm provide more *redundant* information which counts as a part of the target users' attractive information. However, the greedy algorithm misses more other attractive information which is also needed by the target users, resulting in a lower support ratio. The USS algorithm is able to take advantages of the MapReduce information process for the whole user/record dataset to identify a globally optimized connection set, which can supply more attractive information with less spam.

Chapter 7

CONCLUSION

Mobile social networks(MSNs) have emerged as an active and efficient fashion for social network users to make friends, share experience and communicate with each other. As an significant partition of user communication, content dissemination in MSNs has shown its advantages and challenges as well. MSNs mainly connect users through the social relationship, which makes the connections close and greatly inspires users' communication and interaction. On the other hand, users personal interests on the content have significant impact on the user interaction performance. And users' concern on the privacy, efficiency and cost make the content dissemination in MSNs a challenging and meaningful problem. In this dissertation, the content dissemination problem in mobile social networks is studied. By analyzing users interests on the content and corresponding possible behaviors, a series of frameworks and protocols are designed to satisfying users' requirement on content dissemination and enhance the dissemination performance.

The content dissemination for streaming video in MSNs is studied in this dissertation. The requirements and objective for streaming video dissemination is analyzed. To predict the possible behaviors of the social users on video transmission, a Bayesian network based model is derived, which can efficiently analyze the influence of the content, social relationship and physical resources factors.

Another important issue related to content dissemination in MSNs is the requirements from the content. The contents with constraints on the content copy and content reward are studied and analyzed as authorized content, the objective of which is to maximize the reward obtained by content generator. The Maximum Weighted Connected subgraph with node Quota (MWCQ) problem is derived. Two efficient heuristic algorithms, Dynamic Programming based SAID (DP-SAID) and Two-Hop based greedy SAID (THSAID) algorithms,

are derived to provide either accurate or low cost computing solution for the problem. The authorized content dissemination is further studied in Opportunistic Social Networks(OSNs), in which the connections are unstable and unpredictable. The Social Connection Pattern (SCP) is proposed to describe the interest distributions of users social connections. We then develop the Social Connection Pattern based Dissemination (SCPD) algorithm to identify a proper content dissemination strategy when two users contact.

My work on content dissemination in mobile social networks does not only bring contribution on the social communication analysis and dissemination scheme development in MSNs, but also provide certain perspective and guideness for the potential research and development in this area.

REFERENCES

[1] *American Marketing Association*, http://www.ama.org.

[2] *Facebook*, http://facebook.com.

[3] *Twitter*, http://twitter.com.

[4] *Instagram*, http://instagram.com.

[5] *Wi-Fi Alliance*, http://www.wi-fi.org/.

[6] *Wimax*, http://wimaxforum.org.

[7] *LTE*, http://3gpp.org.

[8] Y. Wang and J. Wu, "Social-tie-based information dissemination in mobile opportunistic social networks," in *Proc. of IEEE International Symposium and Workshops on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2013.

[9] F. Santos, B. Ertl, C. Barakat, T. Spyropoulos, and T. Turletti, "Cedo: Content-centric dissemination algorithm for delay-tolerant networks," in *Proc. of MSWiM*, 2013.

[10] Z. Lu, Y. Wen, W. Zhang, Q. Zheng, and G. Cao, "Towards information diffusion in mobile social networks," *IEEE Transactions on Mobile Computing*, vol. 14, pp. 1–13, 2015.

[11] W. Chen and Y. Y. L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *Proc. of ICDM*, 2010.

[12] C. Kong and X. Cao, "Semi-controlled authorized information dissemination in content-based social networks," in *Proc. of ICCCN*, 2014.

[13] W. Gao and G. Cao, "User-centric data dissemination in disruption tolerant networks," in *Proc. of IEEE INFOCOM*, April 2011.

[14] A. J. Mashhadi, S. B. Mokhtar, and L. Capra, "Habit: Leveraging human mobility and social network for efficient content dissemination in delay tolerant networks," in *Proc. of WoWMoM*, 2009.

[15] E. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proc. of Mobihoc*, 2007.

[16] M. Newman, "A measure of betweenness centrality based on random walks," vol. 1, no. 3, 2005.

[17] J. Chen, O. R. Zaiane, and R. Goebel, "Detecting communities in social networks using max-min modularity," in *Proc. of SIAM*, 2009.

[18] A. Pietilänen and C. Diot, "Dissemination in opportsimbetunistic social networks: The role of temporal communities," in *Proc. of the Thirteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2012.

[19] M. Girvan and M. E. J. Newman, "M. girvan and m. e. j. newman," in *Proc.Natl.Acad.Sci.USA*, 2009.

[20] M. Xiao, J. Wu, and L. Huang, "Community-aware opportunistic routing in mobile social networks," *IEEE Trans. on Computers*, vol. 63, no. 7, pp. 1682–1695, 2014.

[21] M. Taghizadeh and S. Biswas, "Community based cooperative content caching in social wireless networks," in *Proc. of Mobihoc*, 2013.

[22] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay tolerant networks," in *Proc. of MobiHoc*, 2007.

[23] Y. Zhang and J. Zhao, "Social network analysis on data diffusion in delay tolerant networks," in *Proc.of Mobihoc*, 2009.

[24] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, no. 3, 2007.

[25] J. Yang, J. McAuley, and J.Leskovec, "Community detection in networks with node attributes," in *Proc. of ICDM*, 2013.

[26] C. boldrini, M. Conti, and A. Passarella, "Contentplace: Social-aware data dissemination in opportunistic networks," in *Proc. of MSWiM*, 2008.

[27] R. Ciobanu, R. Marin, and C. Dobre, "Onside: Socially-aware and interest based dissemination in opportunistic networks," in *Proc. of IEEE Network Operations and Management Symposium (NOMS)*, 2014.

[28] D. Kempe, J. M. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. of KDD*, 2003.

[29] A. Goyal, F. Bonchi, and L. Lakshmanan, "Learning influence probabilities in social networks," in *Proc. of WSDM*, 2010.

[30] A. Mei and J. Stefa, "Give2get: Forwarding in social mobile wireless networks of selfish individuals," in *Proc. of ICDCS*, 2010.

[31] U. B. Shevade, H. H. Song, L. Qiu, and Y. Zhang, "Incentive-aware routing in dtns," in *Proc. of ICNP*, 2008.

[32] T. Ning, Z. Yang, H. Wu, and Z. Han, "Self-interest-driven incentives for ad dissemination in autonomous mobile social networks," in *Proc. of IEEE INFOCOM*, 2003.

[33] V. Schiavoni, E. Riviere, and P. Felber, "Whisper: Middleware for confidential communication in large-scale networks," in *Proc. of IEEE ICDCS*, 2011.

[34] E. Vasserman, R. Jansen, J. Tyra, N. Hopper, and Y. Kim, "Membership-concealing overlay networks," in *Proc. of CCS*, 2009.

[35] A. Singh, G. Urdaneta, M. van Steen, and R. Vitenberg, "Robust overlays for privacy-preserving data dissemination over a social graph," in *Proc. of IEEE ICDCS*, 2012.

[36] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. of KDD*, 2011.

[37] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, and Y. Sun, "Geocommunity-based broadcasting for data dissemination in mobile social networks," *IEEE Trans. on Parallel and Distriubtion Systems*, vol. 24, no. 4, pp. 734–743, 2013.

[38] J. Fan, Y. Du, W. Gao, J. Chen, and Y. Sun, "Geography-aware active data dissemination in mobile social networks," in *Proc. of IEEE Mobile Adhoc and Sensor Systems (MASS)*, 2010.

[39] B. Popescu, B. Crispo, and A. Tanenbaum, "Safe and private data sharing with turtle: Friends team-up and beat the system," *Security Protocols*, vol. 3957, 2006.

[40] W. Gao, G. Cao, M. Srivatsa, and A. Iyengar, "Distributed maintenance of cache freshness in opportunistic mobile networks," in *Proc. of ICDCS*, 2012.

[41] K. Chen and H. Shen, "Global optimization of file availability through replication for efficient file sharing in manets," in *Proc. of ICNP*, 2011.

[42] H. Shuai, D. Yang, W. Cheng, and M. Chen, "Mobiup: An upsampling-based system architecture for high-quality video streaming on mobile devices," *IEEE Trans. on Multimedia*, vol. 13, no. 5, pp. 1077 –1091, 2011.

[43] O. Bonastre and C. Salvador, "A collaborative mobile architecture for multicast live-streaming social networks," in *Proc. of IEEE Int. Conf. on Multimedia and Expo*, 2009.

[44] I. Ullah, G. Bonnet, G. Doyen, and D. Gaiti, "Modeling user behavior in p2p live video streaming systems through a bayesian network," *Lecture Notes in Computer Science*, vol. 6155, pp. 2–13, 2010.

[45] S. Wu, J. Hsu, and C. Chen, "Headlight prefetching and dynamic chaining for cooperative media streaming in mobile environments," *IEEE Trans. on Mobile Computing*, vol. 8, no. 2, pp. 173 –187, 2009.

[46] C. Kong, X. Cao, and M. Liu, "Bayesian-based video sharing in mobile social networks," in *Proc. of IEEE Globecom*, 2013.

[47] Y. Pan, M. Lee, J. Kim, and T. Suda, "An end-to-end multi-path smooth handoff scheme for stream media," in *Proc. of the 1st ACM Int. Workshop on Wireless mobile applications and services on WLAN hotspots*, 2003.

[48] B. Wang, W. Wei, Z. Guo, and D.Towsley, "Multipath live streaming via tcp: Scheme, performance and benefits," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 5, no. 3, pp. 25:1–25:23, 2009.

[49] *OPNET*, http://www.opnet.com/.

[50] L. Strachilevitz, "Social norms from close-knit groups to loose-knit groups," *University of Chicago Law Review*, vol. 70, 2003.

[51] J. Rajahalme, M. S. K. Visala, and J. Riihijärvi, "On name-based inter-domain routing," *Computer Networks*, vol. 55, no. 4, pp. 975–986, 2011.

[52] A. Carzaniga and A. Wolf, "Content-based networking: A new communication infrastructure," in *NSF Workshop on Developing an Infrastructure for Mobile and Wireless Systems*, 2002.

[53] A. Carzaniga, M. Papalini, and A. Wolf, "Content-based publish/subscribe networking and information-centric networking," in *Proc. of ACM SIGCOMM workshop on Information-centric networking*, 2011.

[54] *NFC*, http://nfc-forum.org/.

[55] K. Lin, C. Chen, and C. Chou, "Preference-aware content dissemination in opportunistic mobile social networks," in *Proc. IEEE INFOCOM*, March 2012.

[56] B. Jedari and F. Xia, "A survey on routing and data dissemination in opportunistic mobile social networks," *IEEE Communications Surveys and Totorials*, vol. 99, 2013.

[57] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proc. of ACM IMC*, 2007.

[58] C. Rolim, V. Leithardt, A. Rossetto, T. D. Santos, A. Souza, and C. Geyer, "Six degrees of sepraration to improve routing in opportunistic networks," *Int. Jour. of UbiComp*, vol. 4, no. 3, pp. 11–22, 2013.

[59] T. Hossmann, T. Spyropoulos, and F. Legendre, "A complex network analysis of human mobility," in *Proc. of IEEE Computer Communications Workshops (INFOCOM WKSHPS)*, 2011.

[60] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.

[61] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "Crawdad trace cambridge/haggle/imote/infocom2006," 2009.

[62] A.-K. Pietilainen, "CRAWDAD data set thlab/sigcomm2009 (v. 2012-07-15)," Downloaded from http://crawdad.org/thlab/sigcomm2009/, Jul. 2012.

[63] *PewResearch*, http://www.pewresearch.org/fact-tank/2014/09/24/how-social-media-is-reshaping-news/.

[64] D. Westerman, P. R. Spence, and B. V. D. Heide, "Social media as information source: Recency of updates and credibility of information," *Computer-Mediated Communication*, vol. 19, no. 2, 2013.

[65] J. Naruchitparames, M. Gunes, and S. Louis, "Friend recommendations in social networks using genetic algorithms and network topology," in *Proc. of IEEE Congress on Evolutionary Computation (CEC)*, 2011.

[66] W. Chen, J. Chu, J. Luan, H. Bai, Y. Wang, and E. Chang, "Collabortive filtering for orkut communities: Discovery of user latent behavior," in *Proc. of WWW*, 2009.

[67] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. of WWW*, 2001.

[68] J. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," *The Adaptive Web*, vol. 4321, pp. 291–324, 2007.

[69] Z. Wang, Y. Tan, and M. Zhang, "Graph-based recommendation on social networks," in *Proc. of International Asia-Pacific Web Conference*, 2010.

[70] J. Zhao, J. Lui, D. Towsley, and X. Guan, "Whom to follow: Efficient followee selection for cascading outbreak detection on online social networks," *Computer Networks*, vol. 75, Part B, no. 0, pp. 544 – 559, 2014.

[71] I. Konstas, V. Stathopoulos, and J. Jose, "On social networks and collaborative recommendation," in *Proc. of International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2009.

[72] J. Golbeck, "Generating predictive movie recommendations from trust in social networks," in *Proc. of iTrust*, 2006.

[73] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Procee. of ACM Conference on Recommender Systems*, 2007.

[74] P. Victor, M. D. Cock, and C. Cornelis, "Trust and recommendations," in *Recommender Systems Handbook*. Springer US, 2011, pp. 645–675.

[75] X. Yang, Y. Guo, and Y. Liu, "Bayesian-inference based recommendation in online social networks," pp. 642–651, 2013.

[76] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen, "Lcars: A location-content-aware recommender system," in *Proc. of KDD*, 2013.

[77] V. Zheng, Y. Zheng, X. Xie, and Q. Yang, "Collaborative location and activity recommendations with gps history data," in *Proc. of WWW*, 2010.

[78] Z. Wang, J. Liao, Q. Cao, H. Qi, and Z. Wang, "Friendbook: A semantic-based friend recommendation system for social networks," *IEEE Trans. on Mobile Computing*, vol. 14, no. 3, pp. 538–551, March 2015.

[79] L. Guo, F. You, J. Guo, L. Wu, and X. Zhang, "Sfviz: Interest-based friends exploration and recommendation in social networks," in *Proc. of VINCI*, 2011.

[80] C. Kong, G. Luo, L. Tian, and X. Cao, "Optimizing social connections for efficient information acquisition," in *Proc. of IEEE Globecom*, 2016.

[81] R. Andonov, V. Poirriez, and S. Rajopadhye, "Unbounded knapsack problem: Dynamic programming revisited," *European Journal of Operational Research*, vol. 123, no. 2, pp. 394 – 407, 2000.

[82] *Tweepy*, http://tweepy.org.